



ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE

Fakulta elektrotechnická
Katedra radioelektroniky

Potlačení šumu v barevných videosekvencích

Noise Reduction in Color Video Sequences

Diplomová práce

Studijní program: Komunikace, multimédia a elektronika

Studijní obor: Multimediální technika

Vedoucí práce: Ing. Karel Fliegel, Ph.D.

Jiří Breitkopf

Praha 2018

I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Breitkopf** Jméno: **Jiří** Osobní číslo: **406099**
Fakulta/ústav: **Fakulta elektrotechnická**
Zadávající katedra/ústav: **Katedra radioelektroniky**
Studijní program: **Komunikace, multimédia a elektronika**
Studijní obor: **Multimediální technika**

II. ÚDAJE K DIPLOMOVÉ PRÁCI

Název diplomové práce:

Potlačení šumu v barevných videosekvencích

Název diplomové práce anglicky:

Noise Reduction in Color Video Sequences

Pokyny pro vypracování:

Podejte přehled metod pro efektivní potlačení šumu v barevných videosekvencích. Na základě analýzy vlastností šumu v reálných obrazových datech proveďte simulaci snímacího systému za různých podmínek. Vybrané metody pro potlačení šumu implementujte a ověřte jejich účinnost.

Seznam doporučené literatury:

- [1] Gonzalez, R. C. Digital image processing, Pearson, 2002.
- [2] Bovik, A. Handbook of Image and Video Processing, Elsevier Academic Press, 2005.

Jméno a pracoviště vedoucí(ho) diplomové práce:

Ing. Karel Fliegel, Ph.D., katedra radioelektroniky FEL

Jméno a pracoviště druhé(ho) vedoucí(ho) nebo konzultanta(ky) diplomové práce:

Datum zadání diplomové práce: **13.02.2017** Termín odevzdání diplomové práce: _____

Platnost zadání diplomové práce: **31.08.2018**

Podpis vedoucí(ho) práce

Podpis vedoucí(ho) ústavu/katedry

Podpis děkana(ky)

III. PŘEVZETÍ ZADÁNÍ

Diplomant bere na vědomí, že je povinen vypracovat diplomovou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací. Seznam použité literatury, jiných pramenů a jmen konzultantů je třeba uvést v diplomové práci.

Datum převzetí zadání

Podpis studenta

Prohlášení

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

V Praze dne 8. ledna 2018

.....

Poděkování

Tímto bych zde chtěl poděkovat vedoucímu mé diplomové práce Ing. Karlu Fliegelovi, Ph.D., za jeho vstřícnost a cenné rady, které se pozitivně podílely na celkovém zpracování práce. Dále bych rád poděkoval mé rodině a mým blízkým, kteří mě silně podporovali nejen při vypracovávání této práce, ale i po celou dobu studia.

Abstrakt

Tato diplomová práce se zabývá problematikou potlačení šumu v barevných videosekvencích. Popisuje současné metody a přístupy redukce šumu v obraze. Pomocí analýzy reálného snímacího systému vytváří nekonvenční model šumu, kterým v následných simulacích ověřuje účinnosti vybraných algoritmů pro potlačení šumu za různých podmínek. Jednotlivé algoritmy jsou hodnoceny z hlediska účinnosti redukce šumu pomocí několika objektivních metrik s plnou referencí a také z hlediska jejich výpočetní náročnosti.

Klíčová slova

potlačení šumu, videosekvence, analýza šumu, model šumu, účinnosti redukce šumu

Abstract

This diploma thesis deals with the issue of noise reduction in color video sequences. It describes current methods and approaches to image denoising. Using analysis of a real system, it creates an unconventional noise model, which, in subsequent simulations, verifies the efficiency of selected denoising algorithms under different conditions. Denoising algorithms are rated for noise reduction efficiency using several full reference objective metrics and in terms of their computational demand.

Keywords

noise reduction, video sequences, noise analysis, noise model, noise reduction efficiency

Obsah

Seznam zkratk	10
Úvod	11
1 Typy šumu	12
1.1 Gaussův šum	13
1.2 Impulsní šum	14
1.3 Fotonový šum	15
1.4 Kvantizační šum	15
2 Potlačení šumu	17
2.1 Předpoklady	17
2.2 Metody redukce šumu	17
2.2.1 Lineární metody	18
2.2.2 Nelineární metody	20
3 Redukce šumu pomocí vlnkové transformace	22
3.1 Princip redukce šumu pomocí DWT	24
3.2 Adaptivní volba prahu	26
3.2.1 Visu Shrink	26
3.2.2 Sure Shrink	27
3.2.3 Bayes Shrink	27
4 Videosekvence	28
4.1 Redukce šumu v časové oblasti	28
4.1.1 Odhad a kompenzace pohybu	29
4.1.2 Kálmánova filtrace	29
4.2 Časoprostorová redukce šumu	30
4.2.1 Block Matching 3D Algorithm	30
4.3 Barevný obraz	31
5 Analyzovaný snímací systém	32
5.1 Parametry snímání	32
5.2 Snímací podmínky	33
5.3 Snímaná scéna	33
6 Modelování	34

6.1	Odhad parametrů šumu	34
6.1.1	Analýza signálové závislosti	36
6.1.2	Rozdělení šumu	37
6.1.3	Barevné kanály	39
6.2	Realizace testovacího modelu	40
6.2.1	Tvorba referenční sekvence	40
6.2.2	Modelování šumu	42
6.3	Testované algoritmy	45
6.3.1	Bilaterální filtr	45
6.3.2	Vlnková transformace	45
6.3.3	Kálmánův filtr	46
6.3.4	BM3D a VBM3D	46
6.4	Metriky hodnocení.....	47
6.4.1	MSE	47
6.4.2	PSNR.....	47
6.4.3	SSIM	48
6.4.4	VIF	48
7	Vyhodnocení účinnosti algoritmů	49
7.1	Výsledky objektivního hodnocení redukce šumu	49
7.2	Výpočetní náročnost	64
	Závěr.....	65
	Literatura	67
	Seznam obrázků a tabulek	69
	Přílohy	71

Seznam zkratek

A/D	Analog/Digital (analogový/digitální)
AGC	Auto Gain Control (automatické zesílení)
AVC	Advanced Video Coding (pokročilé kódování videa)
AWGN	Additive White Gaussian Noise (aditivní bílý Gaussovský šum)
BM3D	Block Matching 3D algorithm (blokový porovnávací 3D algoritmus)
CMOS	Complementary Metal-Oxide-Semiconductor
CMY	Cyan, Magenta, Yellow (azurová, purpurová, žlutá)
DCT	Discrete Cosine Transform (diskrétní kosinová transformace)
DWT	Discrete Wavelet Transform (diskrétní vlnková transformace)
GOP	Group of Pictures (skupina obrázků)
IP	Internet Protocol (internetový protokol)
IR	Infra-Red (infračervený)
LAB	Lightness, A and B color opponents (světelnost, kanály určující barvu)
MPEG	Moving Picture Experts Group
MSE	Mean Square Error (střední kvadratická chyba)
nD	n-Dimensional (n-dimenzionální)
PSNR	Peak Signal-to-Noise Ratio (poměr špičkového signálu k šumu)
RAM	Random Access Memory
RGB	Red, Green, Blue (červená, zelená, modrá)
SAE	Sum of Absolute Error (suma absolutních odchylek)
SNR	Signal-to-Noise Ratio (poměr signálu k šumu)
SSD	Solid-Sate-Drive
SSIM	Structural Similarity index (index strukturální podobnosti)
ST-GSM	Spatiotemporal Gaussian Scale Mixture Model
SURE	Stein's Unbiased Risk Estimator
VBM3D	Video Block Matching 3D algorithm
VBM4D	Video Block Matching 4D algorithm
VIF	Visual Information Fidelity (věrnost vizuální informace)
YCbCr	Luminance, Chrominance blue, Chrominance red (jas a chrominanční kanály)
YUV	Luminance, U and V chrominance components (jas a chrominanční kanály)

Úvod

Všechna snímací zařízení přidávají do obrazu jistý druh šumu. Jedná se nejčastěji o rušivé zrnění výsledného videa, které značně snižuje jeho vnímanou kvalitu, například ostrost. Z těchto důvodů je vhodné šum ve videu, i obrazu obecně, potlačit na co možná nejmenší možnou úroveň. Proto v dnešní době existuje řada metod, které se snaží šum odstranit, nebo alespoň výrazně redukovat. Šum vzniká již při samotném exponování snímku, přenosu obrazu, kompresi výsledného videa a dalších operacích.

Co se týče potlačování šumu, existuje velké množství metod i přístupů, jakými se dá šum v obraze redukovat. Jisté rozdíly v metodách, ale spíše v konkrétních přístupech, jsou pro potlačení šumu ve statických fotografiích a videosekvencích, kde se dá využít dalších závislostí, především časové souvislosti jednotlivých snímků ve videu.

V jednotlivých kapitolách teoretické části práce rozebereme základní druhy obrazového šumu, se kterým se můžeme běžně setkat, a popíšeme jednoduché i pokročilejší metody pro jeho potlačení. Zmíníme rozdíly v přístupech potlačování šumu ve statických fotografiích a barevných videosekvencích. Dále v praktické části budeme analyzovat šum u reálné snímací soustavy za různých světelných podmínek. Takto získané vlastnosti šumu využijeme při modelování šumu, kdy takto vytvořeným šumem degradujeme kvalitně nasnímanou sekvenci s minimálním šumem. Pomocí několika metrik a znalosti původního signálu ověříme účinnost vybraných algoritmů pro potlačení šumu. Pro veškerou programátorskou část práce budeme využívat software MATLAB¹.

¹ <https://www.mathworks.com/products/matlab.html>

1 Typy šumu

V této první kapitole teoretické části popíšeme základní druhy obrazového šumu, se kterými se běžně setkáváme při focení, nebo natáčení videa. Uvedeme několik nejčastějších typů šumu a jejich ilustrace.

Faktorů, podílejících se na vzniku šumu, je celá řada a pouze některé z nich máme možnost ovlivnit. Šum se nejčastěji vyskytuje především na snímcích pořízených za špatných světelných podmínek a s užitečným signálem je obvykle silně nekorelovaný. To ale neplatí vždy a některé druhy šumu jsou signálově závislé. Snímací čip funguje na principu převodu energie dopadajícího světla (fotonů) na elektrický náboj. Hodnoty elektrického náboje vygenerované světelným senzorem jsou analogovým signálem, který je následně konvertován převodníkem do výsledného digitálního obrazu. Generovaný elektrický signál může být ovlivněn například jiným elektromagnetickým zářením nebo teplotou polovodičových součástek a integrovaných obvodů. Zde vzniká šum, který se mění v čase i prostoru a v obraze je znatelný jako různobarevné zrnění. To je způsobeno především tím, že pro dostatečnou expozici je třeba vyšší citlivosti a tím pádem i vyšší zesílení signálu. Dále se může jednat i například o nedokonalou strukturu snímacího prvku, kde jsou některé buňky vadné. Takový šum je pak v čase a prostoru neměnný.

Šum v obraze dělíme dle jeho vlastností do několika kategorií [1]. Rozdíly vyplývají již z názvů jednotlivých kategorií. Níže uvedeme základní dělení šumu:

dle vztahu k obrazu

- aditivní
- multiplikativní

dle časové závislosti

- časově závislý (temporal)
- časově nezávislý (fixed-pattern)

Aditivní šum se k užitečnému signálu (v našem případě digitálnímu signálu) v daných místech přičítá. Tedy jednoduše řečeno má každý pixel v obraze o několik hodnot jinou úroveň jasu. Tato změna není závislá na původním signálu. Jedná se o nejčastější modely šumu, které můžeme definovat vztahem [1]:

$$I_n(x, y) = I(x, y) + n(x, y), \quad (1.1)$$

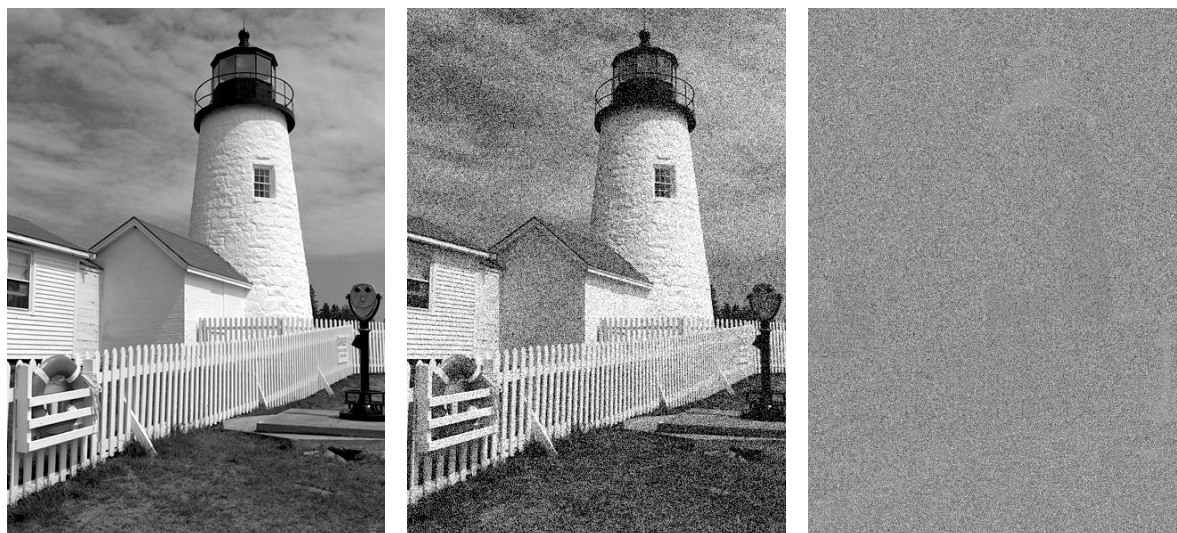
kde I_n je obraz poškozený šumem, I původní obraz a n přidaný šum. **Multiplikativní** šum je oproti šumu aditivnímu závislý na původním signálu, respektive úrovních jednotlivých pixelů v původním obraze, které náhodnou veličinou násobí. Tím pádem je změna hodnot různá v závislosti na původní intenzitě obrazu. **Temporal**, tedy časově závislý šum je takový, jehož hodnoty se velmi rychle a náhodně mění s časem. Kvůli tomu je poměrně složitý jeho odhad a následné úplné odstranění. **Fixed-pattern** (šum s pevným vzorem) je časově nezávislý, v obraze též někdy označován jako prostorově závislý šum. Je neměnný (fixní) v určité

prostorové pozici. Může se jednat například o vadné světlocitlivé buňky na snímači nebo třeba optické vady objektivů. Tento typ je mezi snímky konstantní a není tak složité jeho potlačení ve srovnání s časově závislým šumem.

V dalších podkapitolách jsou stručně rozebrány nejčastější typy šumu, se kterými se běžně setkáváme při práci s obrazovým signálem. Dále také platí, že čím horší snímací zařízení a podmínky pro snímání máme, tím více se šum bude ve výsledném snímku projevovat.

1.1 Gaussův šum

Jedná se o náhodný šum, jehož amplituda má normální (Gaussovo) rozdělení pravděpodobnosti. Nejčastěji se můžeme setkat s aditivním bílým Gaussovským šumem známým jako AWGN (Additive white Gaussian noise), který se používá jako nejbližší aproximační model reálného šumu [1].



Obrázek 1: Ukázka Gaussova šumu (originál, snímek degradovaný šumem, šumové pole)

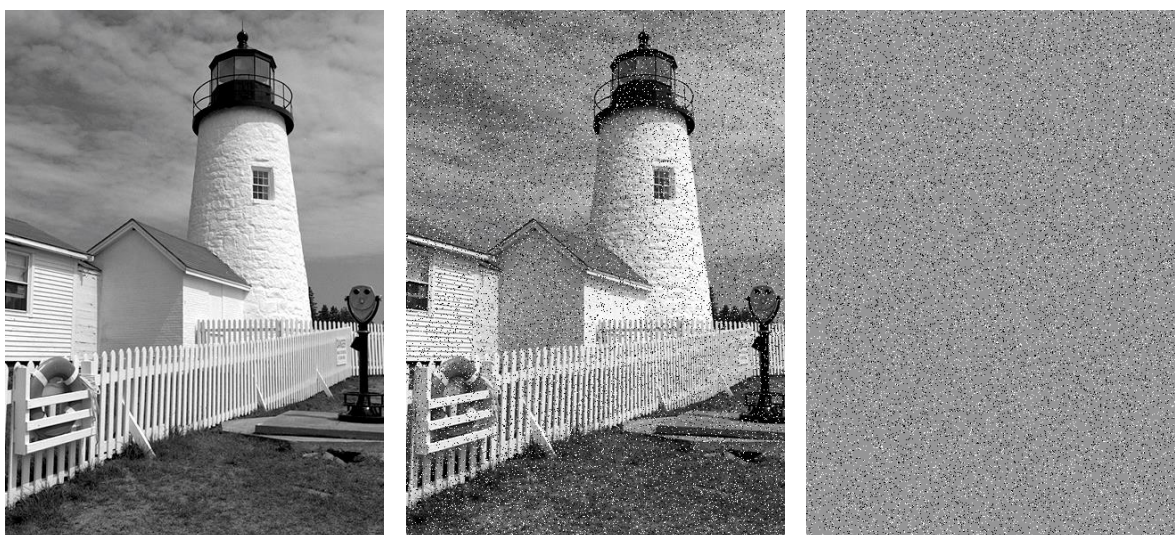
Na výše uvedeném obrázku je ilustrace Gaussova šumu, kdy byl vstupní originální snímek degradován šumem pomocí interní funkce MATLABu *imnoise* typu *gaussian* s nulovou střední hodnotou a rozptylem 0,02. Na obrázku šumového pole (pro lepší viditelnost s roztaženým histogramem) je vidět struktura pozadí snímku v místech, kde dochází k saturaci, jelikož byla zvolena standardní 8bitová reprezentace snímku. Jinak by šumové pole byla pouze matice náhodných hodnot bez viditelné struktury. Tu lze vytvořit například funkcí *normrnd* s odpovídajícími parametry. Následně je možné takto vytvořené šumové pole pomocí definice aditivního šumu k obrázku jednoduše přičíst.

Hustota pravděpodobnosti Gaussovského šumu se střední hodnotou μ a rozptylem σ^2 je dána následujícím vztahem [2]:

$$p_n(x) = \frac{1}{\sigma\sqrt{2\pi}} \cdot e^{-\frac{(x-\mu)^2}{2\sigma^2}}, x \in (-\infty, \infty) \quad (1.2)$$

1.2 Impulsní šum

Impulsní, někdy také nazýván výstřelový šum. Jedná se o případy, kdy jsou postiženy jen ojedinělé body (pixely) v obraze. Takto postižené pixely se vyznačují extrémními hodnotami, čímž silně vybočují z hodnot ostatních pixelů. Některý pixel může být například čistě bílý a některý naopak zase černý. Z tohoto důvodu se takový šum označuje nejčastěji jako šum typu „sůl a pepř“ [1]. Tento typ šumu není příliš složité odstranit pomocí interpolace právě z blízkého okolí bodů kolem zkoumaného bodu pomocí nelineárního mediánového filtru (kapitola 0).



Obrázek 2: Ukázka impulsního šumu (originál, snímek degradovaný šumem, šumové pole)

Obdobný obrázek jako u ilustrace Gaussova šumu můžeme vidět i zde. Šumové pole je zde ilustrováno tak, že šedá hodnota je beze změny, černé a bílé body znázorňují pixely, které mají extrémní hodnoty – bílou, resp. černou. Tento typ šumu je možné opět pomocí funkce *imnoise* typu *salt & pepper* modelovat v MATLABu. Konkrétní ukázka je s hustotou šumu 0,1.

Impulsní šum je možné popsat vztahem [3]:

$$y_{i,j} = \begin{cases} s_{min}, & s \text{ pravd. } p, \\ s_{max}, & s \text{ pravd. } q, \\ x_{i,j}, & s \text{ pravd. } r, \end{cases} \quad (1.3)$$

kde $x_{i,j}$ je obrazový bod na pozici (i,j) původního obrazu, $[s_{min}, s_{max}]$ je dynamický rozsah obrazu, pravděpodobnost r je rovna $1-p-q$ a úroveň šumu je definována jako součet pravděpodobností p a q .

1.3 Fotonový šum

Jedná se o šum buňky, který se projevuje především v systémech, které pracují s malým množstvím energie, respektive světla. Vzhledem k velmi malé aktivní ploše buňky senzoru počítá fotodioda každý foton [4]. Šum je pak z části tvořen i možnou nedokonalostí dané technologie. Příkladem může být světelný, tepelný nebo šum způsobený kvantovými jevy. Obvykle má Poissonovské rozdělení a je částečně korelovaný se signálem. Hustota pravděpodobnosti Poissonova šumu je dána [4]:

$$p_n(k, \lambda) = \frac{\lambda^k e^{-\lambda}}{k!}, \quad (1.4)$$

kde λ je střední hodnota počtu nasčítaných fotonů senzorem v daném časovém intervalu.

1.4 Kvantizační šum

Jedná se o zaokrouhlení hodnoty napětí, které generuje světlocitlivá buňka na nejbližší číslo na stupnici A/D převodníku. Zkrátka převod ze spojitě oblasti do oblasti diskrétních hodnot. Buňky s podobným napětím tak mohou mít jiné číselné hodnoty, jelikož vygenerované napětí padne přímo na rozhraní rozhodovací úrovně. Jedná se vždy o nezvratnou ztrátu informace. Může záviset na signálu, pokud je počet kvantizačních úrovní nízký [5]. Pokud se jedná o lineární A/D převodník (se stejným krokem kvantizačních hladin v celém jeho rozsahu), pak je kvantizační šum modelován jako uniformní s kvantizačním krokem Δ a hustotou pravděpodobnosti popsanou následujícím vztahem [5]:

$$p_n(x) = \begin{cases} \frac{1}{\Delta}, & -\frac{\Delta}{2} \leq n \leq \frac{\Delta}{2}, \\ 0, & \text{jinde.} \end{cases} \quad (1.5)$$

V tomto případě závisí velikost kvantizačního šumu pouze na počtu kvantizačních úrovní, jelikož kvantizační chyba je u všech lineárních převodníků stejná ($-1/2$ až $+1/2$ kvantizační

úrovně). Velikost šumu se obvykle vyjadřuje jako poměr užitečného signálu ku šumu v decibelech [5].

$$SNR = 10 \cdot \log \left(\frac{P_{sig}}{P_{noise}} \right) [dB], \quad (1.6)$$

kde P_{sig} je výkon signálu a P_{noise} výkon šumu. Odstup signálu od kvantizačního šumu je pak dán následujícím vztahem:

$$SNR_{A/D} = 20 \cdot \log 2^N \approx 6,02 \cdot N [dB], \quad (1.7)$$

kde N je počet kvantizačních úrovní převodníku.

Na následujícím obrázku je ilustrováno, jak vypadá obrázek s vysokým kvantizačním šumem. Původní obrázek o 256 úrovních budeme v naší ilustraci chápat jako spojitý signál. Degradovaný snímek je pak původní, silně podkvantovaný obrázek. Následně je zobrazen kvantizační šum, tedy rozdíl vstupního originálu a výstupního podkvantovaného snímku.



Obrázek 3: Ilustrace kvantizačního šumu

V této kapitole jsme se zabývali pouze jednoduchými modely šumu v šedotónových obrázcích pro jejich lepší názornost a ukázkou. Šum u reálné snímací soustavy je pak mnohem komplexnější a často se skládá z několika různých výše uvedených základních typů šumu. Videosekvencemi a následnou analýzou šumu u reálné snímací soustavy se pak zabýváme v kapitolách 4 a 6.1.

2 Potlačení šumu

Vysoká úroveň šumu v obraze je velmi nežádoucí. Může silně omezovat jak rozlišení jednotlivých detailů, snížit ostrost hran i celkový kontrast snímku, tak i poškodit vnímání jednotlivých ploch a přechodů. Proto, pokud je snímek postižen šumem, je vhodné jeho intenzitu alespoň snížit, pokud nelze ze snímku zcela odstranit. K tomu by byla nutná znalost šumu nebo původního signálu, které pochopitelně není možné přesně získat. Proto je nutný co možná nejpřesnější odhad parametrů šumu a druh, kterým původní snímek degradoval.

V následujících podkapitolách rozebíráme základní předpoklady, které by měl šum splňovat, aby byl možný jeho odhad a rozlišení od původního signálu, a následně základní metody, zabývající se potlačením daných typů šumu.

2.1 Předpoklady

Aby bylo možné nežádoucí šum v obraze potlačit, je nutná znalost nebo alespoň přibližný odhad jeho parametrů a druh přidaného šumu. To ale není vždy jednoduché určit. Celkově rozlišit od sebe skutečný užitečný signál a nežádoucí přidaný šum je vcelku složité. Měl by ale zpravidla splňovat následující aspekty [1].

- Náhodná veličina
- Nízká korelace s původním signálem
- Nulová střední hodnota
- Nízká energie vůči původnímu signálu

2.2 Metody redukce šumu

Existuje několik metod pro odstranění nebo spíše potlačení šumu v obraze. Každá metoda potlačuje jiný druh šumu. Je tedy důležité zvolit vhodnou metodu pro daný šum s ohledem na to, aby zůstal zachován užitečný signál v co nejlepší možné podobě. Metody můžeme v principu rozdělit na metody lineární a metody nelineární.

2.2.1 Lineární metody

Pro běžný aditivní šum by se mohla zdát dostatečná **konvoluční filtrace**. Tato metoda patří mezi nejjednodušší metody pro potlačení šumu. Funguje na principu odfiltrování kmitočtových složek mimo pásmo s užitečným signálem. Pomocí konvolučního jádra (masky) se nová hodnota pixelu získává například jako průměr okolních pixelů, čímž se sice docílí redukce šumu, ale dojde také vlivem průměrování k redukci prostorové rozlišovací schopnosti, tj. rozostření snímku, což je nežádoucí.



Obrázek 4: Ukázka konvoluční filtrace

Zde je asi nejpoužívanější **Gaussův filtr** [6]. Ilustrace (Obrázek 4) je demonstrována za použití funkce *imgaussfilt* s parametrem směrodatné odchylky sigma 3. Zde se oproti předchozímu případu prostého průměrujícího filtru počítá vážený průměr hodnot v daném okolí pixelu. Vyšší váhy mají pixely, které jsou blízko aktuálně počítaného pixelu a s rostoucí vzdáleností od něj pak váhy klesají. Váhy pixelů v konvoluční masce jsou dány Gaussovou funkcí G:

$$G(x, y) = \frac{1}{2\pi\sigma^2} \cdot e^{-\frac{x^2+y^2}{2\sigma^2}}, \quad (2.1)$$

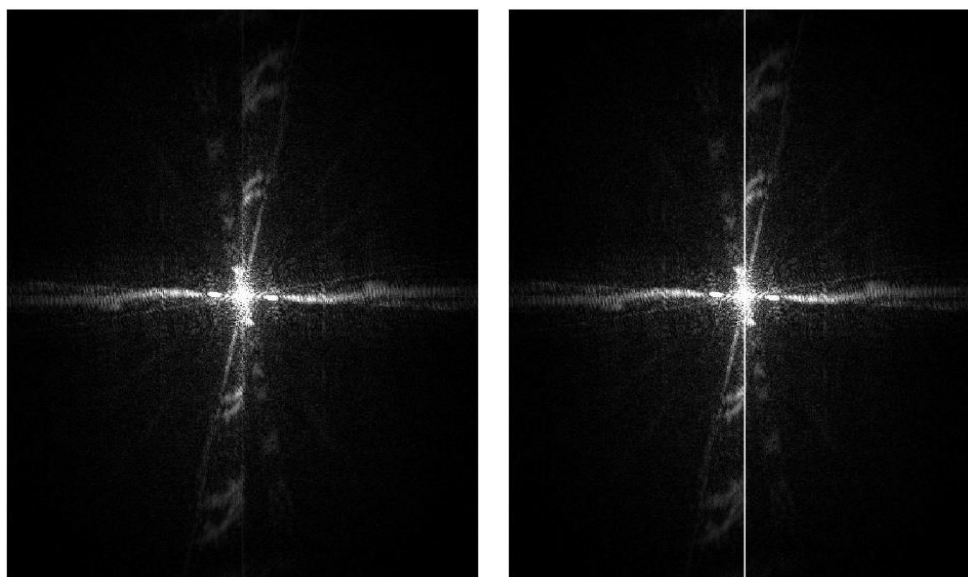
kde x a y udávají vzdálenost a σ udává rychlost klesání funkce v závislosti na vzdálenosti.

S rostoucím σ budou mít vzdálené body menší váhy. Aproximace Gaussova filtru pro $\sigma = 1$ o velikosti 5 x 5 obrazových bodů je uvedena na obrázku níže.

$$\frac{1}{273} \cdot \begin{array}{|c|c|c|c|c|} \hline 1 & 4 & 7 & 4 & 1 \\ \hline 4 & 16 & 26 & 16 & 4 \\ \hline 7 & 26 & 41 & 26 & 7 \\ \hline 4 & 16 & 26 & 16 & 4 \\ \hline 1 & 4 & 7 & 4 & 1 \\ \hline \end{array}$$

Obrázek 5: Příklad Gaussova filtru

Maskování ve spektrální oblasti je další velmi jednoduchou metodou, která je vhodná především pro potlačení úzkopásmového šumu. Ten se ve spektru signálu projevuje jako čára nebo bod. Maskováním spektra můžeme realizovat různé druhy filtrů. Na obrázku níže je uvedena ukázka úzkopásmového šumu, který se ve spektru projevil jako svíslá čára. Za pomoci MATLABu ho bylo velmi jednoduché odstranit, kdy pomocí funkce *fftshift* získáme obrázek ve spekterální oblasti (Obrázek 6), kde prostým nahrazením hodnot v daném místě průměrem z okolních bodů dojde k téměř absolutnímu potlačení šumu v obraze. Výsledek pak získáme zpětným převodem ze spektra pomocí funkce *ifftshift*. Další možností je realizace **dolní propusti**, jelikož se předpokládá, že šum je zpravidla vysokofrekvenční.



Obrázek 6: Úzkopásmový šum ve spektru

2.2.2 Nelineární metody

Další třídou jsou nelineární metody, resp. filtry, které se od metod lineárních liší především v tom, že jejich výstupní hodnoty nejsou stanoveny jako lineární kombinace vstupních hodnot, jako je tomu právě u metod lineárních. Zde se výstupní hodnoty určují dle jiného algoritmu, který například vybírá některou z hodnot z nastaveného okolí filtrovaného bodu. Níže je uvedeno několik používaných metod od základní mediánové filtrace až po pokročilejší metody, jako je například bilaterální filtrace.

Mediánová filtrace (v MATLABu funkce *medfilt2*) je metoda vhodná k odstranění impulsního šumu typu „sůl a pepř“. Jedná se o nelineární metodu, kdy je pixel určen mediánem, tedy hodnotou ležící uprostřed seřazené posloupnosti hodnot, z okolních pixelů ve zvoleném okolí [1]. Tato metoda je principiálně podobná konvoluční filtraci, nicméně se nevyužívá externí matice (konvoluční jádro) pro násobení koeficientů, ale určuje se přímo z nejbližších sousedů daného pixelu. Oproti konvoluční filtraci také lépe zachovává hrany.

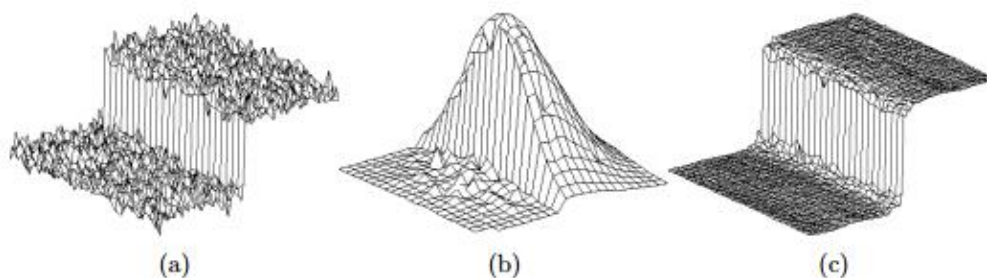
Podobná metoda pro odstranění izolovaných pixelů s výjimečně vysokou nebo naopak nízkou intenzitou je tzv. **outlier filtr**. Tato metoda filtrace nejprve vyhledává špatný obrazový bod. Například z blízkého okolí zjišťuje průměrnou intenzitu a následně porovnává, o kolik se hodnota liší. Pokud je rozdíl větší než stanovený povolený limit, je tento pixel detekován jako vadný a jeho hodnota je nahrazena vybranou hodnotou [7]. To může být zase například průměr nebo medián. Pokud není pixel vyhodnocen jako vadný, jeho hodnota zůstává beze změny. Metoda na tomto principu se někdy označuje jako **konzervativní vyhlazení**. Metoda je rychlá a nenáročná.

Bilaterální filtrace je jednou z dalších již pokročilejších metod. Na rozdíl od předešlých metod kombinuje závislost jak na blízkém okolí od zpracovávaného pixelu (Gaussův filtr), tak na vzdálenosti, respektive podobnosti jejich hodnot. Tím umožňuje dát pixelům, které mají výrazně jinou hodnotu od referenčního pixelu, malou (v limitním případě až nulovou) váhu a brát tedy v potaz pouze ty pixely, jejichž hodnoty jsou blízké právě zpracovávanému pixelu. To znamená, že se při filtraci vyhlazují pouze oblasti s podobnými hodnotami (barevné plochy) a hrany v obraze zůstávají zachovány. Nevýhoda tohoto filtru je v jeho vysoké výpočetní náročnosti [6].

Bilaterální filtr je definován jako:

$$I^B(x) = \frac{\sum_{y \in N(x)} I(y) g_{\sigma_S}(\|I(y) - I(x)\|) g_{\sigma_R}(\|y - x\|)}{\sum_{y \in N(x)} g_{\sigma_S}(\|I(y) - I(x)\|) g_{\sigma_R}(\|y - x\|)}, \quad (2.2)$$

kde g_{σ_S} a g_{σ_R} jsou jádra Gaussova filtru, $N(x)$ je množina pixelů okolo referenčního pixelu a $I(x)$ značí jeho intenzitu. g_{σ_S} ovlivňuje míru vyhlazení a g_{σ_R} míru zachování hran.



Obrázek 7: Bilaterální filtr: (a) původní signál s hranou, (b) jádro filtru, (c) výsledek filtrace
(Převzato z [6])

Mezi další pokročilé metody určitě patří také technika **non-local means**. Na rozdíl od zatím popisovaných metod, které určují hodnotu aktuálně zpracovávaného pixelu z jeho blízkého okolí, metoda non-local means tuto váženou hodnotu určuje z většího okolí, až ze všech obrazových bodů. Váha je závislá pouze na podobnosti se zpracovávaným pixelem [6].

Velmi účinnou metodou pro potlačení šumu je také využití **vlnkové transformace** [8] s vhodným prahováním detailních vlnkových koeficientů, které se budeme podrobněji věnovat v kapitole 3.

3 Redukce šumu pomocí vlnkové transformace

Vlnková transformace je integrální transformace, která umožňuje získat časově-frekvenční (v případě obrázků pak prostorově-frekvenční) popis signálu. Slouží k dekorelaci vstupních dat na nezávislé komponenty a uplatňuje se například při detekci nespojitostí signálu a jeho derivací, identifikaci okamžitých frekvencí, extrakci příznaků nebo kompresi signálů. Je ale vhodná také k potlačení šumu [6].

Na rozdíl od Fourierovy transformace není vstupní obraz rozkládán do harmonických funkcí ale do vlnek, jež mohou mít libovolnou podobu, ale musí však splňovat určité podmínky [9]. Vlnky mění jak svoji velikost, tak i polohu. Díky tomu lze nejen získat informace o obsažené frekvenci v signále, ale i její polohu. Spojitá vlnková transformace je definována takto [10],

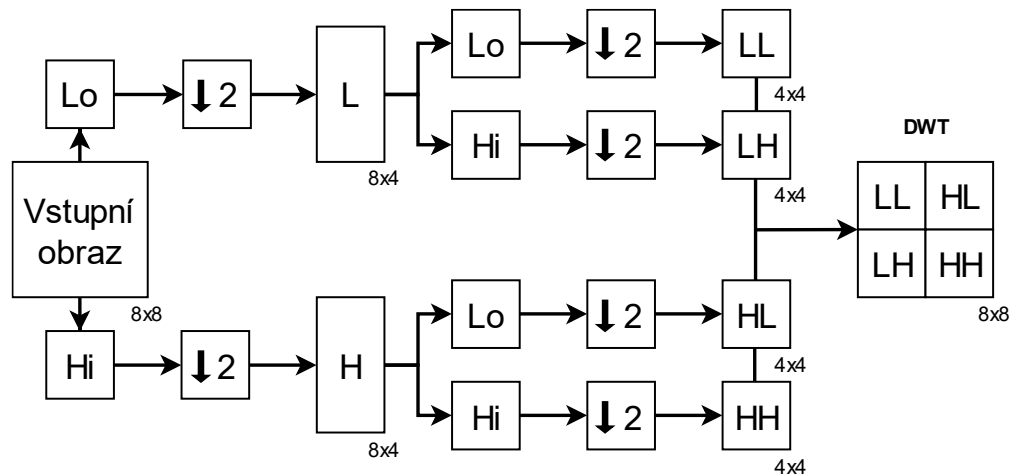
$$W_f(\tau, s) = \int_{-\infty}^{+\infty} f(t) \cdot \overline{\psi_{\tau,s}(t)} \cdot dt, \quad (3.1)$$

kde $\psi_{\tau,s}(t)$ je vlnka, proměnná s je měřítko a proměnná τ představuje posun po ose t . Pro vlnku můžeme uvést vztah,

$$\psi_{\tau,s}(t) = \frac{1}{\sqrt{s}} \cdot \psi\left(\frac{t - \tau}{s}\right), \quad (3.2)$$

kde $s, \tau \in \mathbb{R}, s \neq 0$. Převrácená hodnota odmocniny s slouží k normalizaci energie při změnách měřítka a ψ je mateřská vlnka. Signál je postupně porovnáván s vlnkou, která je posunována v čase. Na ose x nemusí být jen čas. V případě obrazu se může jednat například o polohu. Protože by spojitá vlnková transformace produkovala nekonečné množství vlnkových koeficientů, vznikla také její diskrétní podoba, kdy proměnné s a τ mohou nabývat jen určitých hodnot. Existuje velké množství běžně používaných mateřských vlnek, kdy má většina vlnek poměrně malou oblast nevhodnějšího použití. Proto je vhodný výběr mateřské vlnky velmi důležitý.

Na následujícím obrázku je popsán postup dekompozice obrázku pomocí dvourozměrné diskrétní vlnkové transformace [9]. Bloky H_i resp. L_o představují impulsní odezvu horní propusti resp. dolní propusti. Bloky $\downarrow 2$ představují podvzorkování signálu na polovinu.



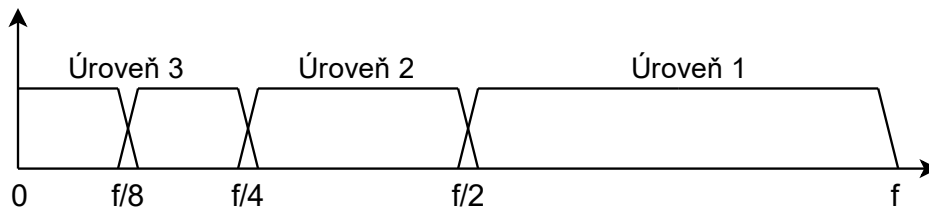
Obrázek 8: 2D DWT dekompozice

Nejprve je aplikován filtr dolní a horní propusti po řádcích. Výstup je decimován s řádem 2. Na každý z těchto vzniklých snímků je opětovně aplikován filtr dolní a horní propusti, tentokrát ale po sloupcích a výstup je opět decimován s řádem 2. Získáme tím čtyři sady koeficientů (4 obrázky) obsahující aproximaci původního obrázku (LL), detaily v horizontálním směru (LH), detaily ve vertikálním směru (HL) a detaily ve vertikálním směru (HH). Tento postup se někdy nazývá jako dyadická dekompozice. Výsledný obraz pak vypadá následujícím způsobem.



Obrázek 9: Rozklad obrázku pomocí DWT

Je možné provést několik stupňů dekompozice. Každý další stupeň dekompozice extrahuje ze signálu různě podrobné detaily. To je nutné, aby mohl být signál analyzován ve více frekvenčních pásmech. Po prvním rozkladu signálu se následně pracuje s aproximačními koeficienty. Ty se podrobují další filtraci, stejně jako v předchozím kroku. Zpravidla se používají 3, případně 5 dekompozičních stupňů. Ve frekvenční oblasti pak vypadá rozklad tímto způsobem [9].



Obrázek 10: Dekompozice ve frekvenční oblasti

3.1 Princip redukce šumu pomocí DWT

Redukce šumu a vyhlazení signálu pomocí diskretní vlnkové transformace se provádí například pomocí vhodného prahování detailních vlnkových koeficientů na dané dekompoziční hladině. Šum se totiž obecně vyskytuje převážně na vyšších frekvencích. Po aplikaci vlnkové transformace získáme vstupní signál rozdělený na detailní a aproximační koeficienty, což je velmi vhodné. Šum se pak díky jeho vlastnostem vyskytuje především v detailních koeficientech, které pak můžeme vhodně redukovat. Tím se také postupně zbavujeme šumu. Pro prahování koeficientů je možné použít různé postupy. Nejčastěji se ale jedná o tvrdé (hard) nebo měkké (soft) prahování [9].

Při použití tvrdého prahování se koeficienty, které jsou menší nebo rovny nastavenému prahu nahrazují nulami. Ostatní hodnoty zůstávají stejné

$$\rho_{\lambda}^{hard}(x) = \begin{cases} x & |x| \geq \lambda \\ 0 & |x| < \lambda \end{cases} \quad (3.3)$$

naproti tomu při použití měkkého prahování se navíc nezměněné koeficienty z tvrdého prahování posouvají o velikost prahu směrem k nule.

$$\rho_{\lambda}^{soft}(x) = \begin{cases} sgn(x) \cdot (|x| - \lambda) & |x| \geq \lambda \\ 0 & |x| < \lambda \end{cases} \quad (3.4)$$

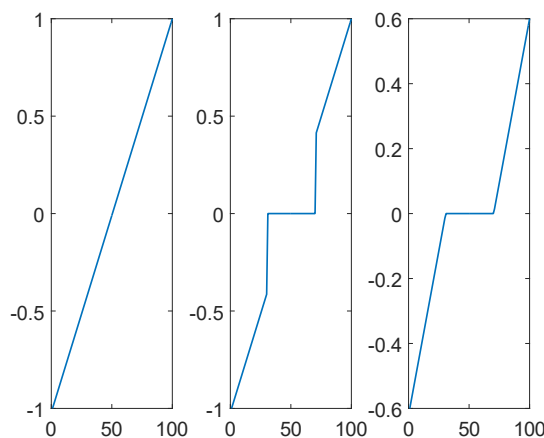
Volba prahu závisí na použitých vlnkách, ale i na charakteru šumu. Může být odlišná v různých měřítkách a pro normální rozdělení šumu se používá tzv. univerzální práh z rovnice uvedené níže [10]:

$$\lambda = \sigma \cdot \sqrt{2 \cdot \ln(M \cdot N)} \quad (3.5)$$

kde M a N představují rozměr obrazové matice a σ směrodatnou odchylku šumu. Tento vztah předpokládá, že velikost šumu v signálu bude pravděpodobně menší než jeho hodnota. To vychází z následujícího tvrzení, kdy z_i je posloupnost nezávislých náhodných veličin s normálním rozdělením $N(0, \sigma^2)$, pak platí [11]

$$\lim_{n \rightarrow \infty} P\left(\max_i |z_i| > \sigma \cdot \sqrt{2 \cdot \ln(n)}\right) \rightarrow 0. \quad (3.6)$$

Práh nemusí být pro každou úroveň dekompozice stejný a při jeho určování musíme brát v úvahu, jakým šumem je snímek poškozen. Tento fakt je reprezentován směrodatnou odchylkou šumu σ . Dále je nutné, stejně jako při použití běžných metod redukce šumu, volit jakýsi kompromis mezi intenzitou potlačení šumu a přílišnou ztrátou detailní informace v obraze. Při použití příliš velkého prahu bychom odstranili nebo modifikovali velké množství vlnkových koeficientů, což by mělo za následek přílišné vyhlazení obrazu.



Obrázek 11: Ilustrace prahování: originál, tvrdé prahování, měkké prahování

Po zvoleném prahování koeficientů se signál opět složí do původní podoby aplikací inverzní diskrétní vlnkové transformace. Postup je obdobný jako při dekompozici signálu. Nejprve je nutné podvzorkované koeficienty transformace zpětně převzorkovat na původní velikost. To se provádí vložením nulových hodnot mezi vzorky. Výsledný signál je posléze filtrován odpovídajícími rekonstrukčními filtry.

3.2 Adaptivní volba prahu

V tomto odstavci jsou dále stručně popsány pokročilejší metody prahování. Jedná se o adaptivní metody, kdy se hodnota prahu určuje zvlášť pro každé subpásmo. Tedy jinými slovy se jedná o metody, které určují nejvhodnější možný práh pro daný obraz v závislosti na tom, jaký šum obsahuje.

Konkrétně se jedná o metody:

- Visu Shrink [12]
- Sure Shrink [13]
- Bayes Shrink [14]

3.2.1 Visu Shrink

První popisovanou adaptivní, Visu Shrink. Jedná se o metodu, která byla prvně představena dvojicí matematiků Donohem a Johnstonem [12]. Tato metoda si dokáže poradit pouze s aditivním šumem, což můžeme považovat za její nevýhodu. U této metody je nutné nejprve odhadnout směrodatnou odchylku šumu σ , s využitím detailních koeficientů vlnkové transformace ze vztahu

$$\hat{\sigma} = \frac{\text{median}(|g_{j-1,k}| : k = 0, 1, \dots, 2^{j-1} - 1)}{0,6745}, \quad (3.7)$$

kde $g_{j-1,k}$ představuje detailní koeficienty vlnkové transformace. Je vhodné využití právě detailních koeficientů, jelikož v nich bývá nejvíce zastoupen přidaný šum. Tento odhad směrodatné odchylky se pak využije ve vztahu univerzálního prahu (3.5), díky čemuž se určí práh právě pro metodu Visu Shrink. Konstanta 0,6745 vychází z předpokladu $E(\text{median}(|u_n|)_{0 \leq n < N}) \approx 0,6745\sigma$, kdy $\{u_n\}$ je posloupnost N nezávislých hodnot s Gaussovským rozdělením, nulovou střední hodnotou a rozptylem σ^2 [9].

3.2.2 Sure Shrink

Další, ještě pokročilejší metodou je Sure Shrink. Jedná se o metodu podobnou metodě Visu Shrink s tím rozdílem, že se práh určuje pro každou úroveň dekompozice signálu zvlášť. Kombinuje se zde univerzální práh a SURE (Stein's Unbiased Risk Estimator) práh [13]. Hodnota prahu metody Sure Shrink je definována jako

$$\lambda_{SURE} = \min \left(\lambda, \sigma \sqrt{2 \cdot \ln(M \cdot N)} \right), \quad (3.8)$$

kde λ je hodnota univerzálního prahu, σ je směrodatná odchylka šumu a M a N rozměry obrazové matice.

3.2.3 Bayes Shrink

Model této metody je vyjádřen jako součet vlnkové transformace původního obrazu s vlnkovou transformací částí kontaminovaných šumem s normálním rozdělením [14]. Jelikož jsou tyto členy na sobě nezávislé, jsou nezávislé i jejich směrodatné odchylky, které můžeme vyjádřit následujícím vztahem

$$\sigma_y^2 = \sigma_x^2 + \sigma_n^2, \quad (3.9)$$

kde σ_y^2 je směrodatná odchylka vlnkové transformace signálu kontaminovaného šumem. Směrodatná odchylka šumu $\hat{\sigma}_n^2$ může být odhadnuta z první [15] dekompoziční úrovně vlnkových koeficientů HH jako

$$\hat{\sigma}_n^2 = \left[\frac{\text{median}(|HH_1|)}{0,6745} \right]^2 \quad (3.10)$$

Výsledný práh je obdobně jako metoda Sure Shrink určován pro každé subpásmo signálu zvlášť. Popsán je pak následujícím vztahem

$$\lambda_{BAYES} = \begin{cases} \frac{\hat{\sigma}_n^2}{\hat{\sigma}_x}, & \text{pro } \hat{\sigma}_n^2 < \hat{\sigma}_y^2 \\ \max\{|A_m|\}, & \text{jinde,} \end{cases} \quad (3.11)$$

kde $\hat{\sigma}_x = \sqrt{\max(\hat{\sigma}_y^2 - \hat{\sigma}_n^2, 0)}$ a A_m jsou vlnkové koeficienty daného subpásma, v jakém je práh určován.

4 Videosekvence

Zatím jsme řešili pouze problematiku ve statických snímcích. Všechny dříve zmíněné metody redukce šumu lze pochopitelně aplikovat i na videosekvenci. V tomto případě by ale z hlediska účinnosti potlačení šumu nebylo vhodné aplikovat jednotlivé filtry a metody na každý jednotlivý snímek bez vzájemné souvislosti mezi nimi. Je totiž obecně známo, že mezi jednotlivými snímky je vysoká míra korelace, tedy vzájemné podobnosti. Díky tomuto faktu můžeme využít metody, které zohledňují časovou souvislost jednotlivých snímků například průměrováním v čase u statických scén, ale také například odhadu pohybu s pohybovou kompenzací mezi jednotlivými snímky u dynamických scén.

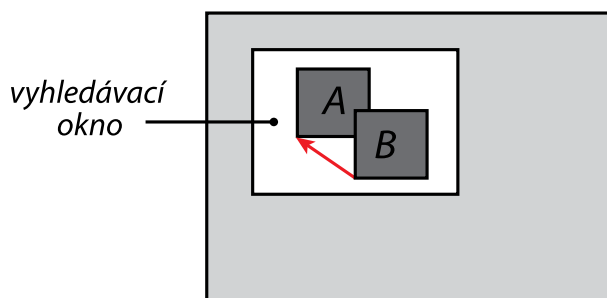
V zásadě tedy existují tři možné přístupy potlačení šumu ve videosekvenci. Buďto přístup založený pouze na prostorové redukci šumu v obraze, kdy je na video nahlíženo jako na jednotlivé nezávislé snímky, ve kterých se potlačuje šum individuálně. Tyto metody byly již popsány, a proto se jim zde nebudu více věnovat. Dále přístup využívající časové závislosti jednotlivých snímků, kde vzhledem k tomu, že se vždy nejedná o naprosto statickou scénu, se také využívají již výše zmíněné algoritmy pohybové kompenzace mezi snímky. A poslední možností jsou metody využívající jak prostorové, tak časové závislosti k potlačení šumu. Tyto metody bývají obvykle nazývány jako 3D redukce šumu [16].

4.1 Redukce šumu v časové oblasti

Metody potlačení šumu ve videosekvenci založené na vzájemné časové souvislosti jednotlivých snímků sekvence jsou velmi efektivní. Může se jednat o prosté časové průměrování jednotlivých pixelů, nebo třeba adaptivní změnu parametrů algoritmů na potlačení šumu v závislosti na vývoji sekvence. Obyčejné časové průměrování je vhodné využít například u statických scén, a to především v místech neobsahujících pohyb, kde jsou změny hodnot pixelů způsobeny především pouze přidáním náhodné veličiny šumu. Příkladem zde může být záznam ze statických průmyslových nebo bezpečnostních kamer. Časově proměnný šum se pak projevuje jako jistě všem dobře známé zrnění v obraze. Pomocí průměru z několika snímků se toto zrnění značně potlačí a samotný výpočet těchto hodnot je velmi jednoduchý a nenáročný. Avšak pokud je ve scéně pohyb, není možné tento jednoduchý přístup využít, protože veškeré pohybuující se objekty za sebou začnou vlivem průměrování zanechávat jakousi stopu, příznačně označovanou ghosting, což je velmi rušivé. Je tedy vhodné filtr nastavit tak, aby při detekci pohybu přestal dané pixely v čase průměrovat, případně při průměrování upravil průměrující okno tak, aby nastavil v čase starším snímkům nižší váhu. Dalším vylepšením je zde využití predikce pohybu, kdy se budou dané pixely průměrovat s pixely na jiných místech vzhledem k jejich budoucímu pohybu. To je ale jen jeden ze základních možných přístupů, který při vhodném nastavení a použití poskytuje velmi dobré výsledky jak z hlediska účinnosti, tak výpočetní náročnosti.

4.1.1 Odhad a kompenzace pohybu

K vylepšení metod využívajících časové souvislosti jednotlivých snímků u videosekvencí jsou důležité také algoritmy pro detekci a kompenzaci pohybu objektů (bloku pixelů) ve scéně. Asi nejrozšířenějším algoritmem je zde blokový porovnávací (**block matching**) algoritmus. Jedná se o algoritmus založený na vcelku jednoduchém principu porovnávání jednotlivých bloků obrazu, který je například součástí kodeku MPEG, který je znázorněn na následujícím obrázku. Nejčastěji se používají čtvercové bloky o velikosti 8×8 nebo 16×16 pixelů. Při použití menších bloků by rostla výpočetní náročnost, naopak při použití větších bloků by silně rostla chyba odhadu. V čase t je blok A porovnán se snímkem v čase $t-1$. Pro nalezení nejpodobnějšího bloku ve snímku se z důvodů snížení výpočetní náročnosti a předpokladu, že se objekt neposune příliš daleko, využívají tzv. vyhledávací okna. Ty udávají, které nejbližší okolí se bude prohledávat. Po nalezení odpovídajícího bloku (blok B) je vytvořen vektor pohybu (červená šipka) pro daný blok pixelů, který určuje směr a velikost pohybu daného bloku. K porovnání jednotlivých bloků slouží několik metrik, nejčastěji se však používá suma absolutních odchylek **SAE**. Ve snímku v čase t je pak využit blok B z předešlého snímku, u kterého se jenom změní jeho pozice.



Obrázek 12: Ilustrace pohybové kompenzace

4.1.2 Kálmánova filtrace

Jedná se o iterativní algoritmus pracující v časové oblasti, který predikuje neznámou hodnotu pixelu na dané prostorové pozici z předchozího pozorování. Může pracovat v reálném čase a s rostoucím počtem snímků se jeho odhad zpřesňuje. Odhady vytvořené z předchozího pozorování totiž porovnává s aktuálním vstupem a dle toho svůj predikční model dále zpřesňuje. Na rozdíl od pouhého průměrování snímků skrze časovou osu se dokáže lépe adaptovat na změny ve scéně. Výhodou tohoto algoritmu je stejně jako u časového průměrování, že nemusíme mít žádnou předchozí znalost o typu daného rušení nebo o samotném původním signálu. Korekce Kálmánova filtru se obecně řídí rovnicemi na další straně [17].

$$KG_t = \frac{E_{\hat{I}_{t-1}}}{E_{\hat{I}_{t-1}} + E_I}, \quad (4.1)$$

$$\hat{I}_t = \hat{I}_{t-1} + KG_t \cdot (I_t - \hat{I}_{t-1}), \quad (4.2)$$

$$E_{\hat{I}_t} = (1 - KG_t) \cdot E_{\hat{I}_{t-1}}, \quad (4.3)$$

kde KG je zisk Kálmánova filtru, E_I je chyba měření (v našem případě odhad rozptylu šumu), $E_{\hat{I}}$ je chyba predikce a \hat{I}_t je predikovaná hodnota v čase t .

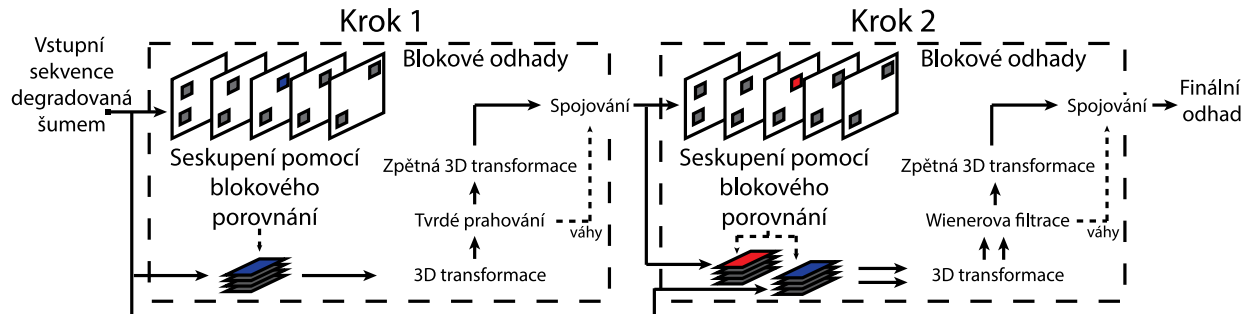
4.2 Časoprostorová redukce šumu

Mezi nejpokročilejší přístupy pro redukci šumu u videosekvencí patří již zmiňované metody využívající časoprostorové souvislosti jednotlivých snímků. Je zde několik velmi pokročilých algoritmů, které efektivně potlačují nežádoucí šum. Jedná se již o vcelku dosti složité metody kombinující různé, výše popisované filtrace nebo algoritmy. Jen pro ukázkou můžeme uvést algoritmus **ST-GSM** (Spatiotemporal Gaussian Scale Mixture Model) a asi v současné době nejpokročilejší **BM3D** algoritmus, který podrobněji popisují v následující podkapitole. Informace o principech ST-GSM algoritmu je v odpovídající literatuře [18].

4.2.1 Block Matching 3D Algorithm

Asi nejrozšířenějším a nejpoužívanějším algoritmem, kombinujícím přístup jak časového, tak prostorového průměrování je Block Matching 3D Algorithm [19]. Jedná se o pokročilý blokový porovnávací algoritmus pro potlačení šumu. Skládá se ze tří dílčích částí. Seskupení dle blokového porovnání (Grouping), společné filtrování (Collaborative filtering) a výsledného spojování (Aggregation). V první části – seskupení, se využívá koncept dvourozměrného blokového porovnání k vyhledání podobných vzorů pro každý referenční vzor, které se následně seskupí do 3D polí. Při využití u videosekvencí se algoritmus obvykle označuje jako **VBM3D** a využívá také predikce v časovém okně pro vyhledání podobných vzorů. Společné filtrování je pak hlavní částí algoritmu pro potlačení šumu. Využívá se zde vzájemné podobnosti mezi jednotlivými vzory z předchozího kroku s aplikací 3D transformace. Ta vykazuje rozptýlenější reprezentaci jednotlivých vzorů než 2D transformace. Následně je využito tvrdého prahování k potlačení šumu a aplikace inverzní 3D transformace. V poslední části spojování se vzory s potlačeným šumem zpětně poskládají na svá původní místa a u pixelů, které se překrývají, se vypočte jejich vážený průměr [20]. VBM3D má dva kroky, kdy v druhém kroku využívá odhadu z prvního a Wienerovy filtrace k dalšímu zpřesnění finálního odhadu. Tento postup můžeme vidět na schématu na další straně (Obrázek 13).

Ještě o něco pokročilejší je **VBM4D**. Jedná se řekněme o rozšíření VBM3D algoritmu, jelikož neseskupuje podobné bloky, ale vzájemně podobné svazky 3D bloků pomocí 4D transformace. Díky tomu je dosaženo ještě vyšší účinnosti redukce šumu, ale také je tím podstatně zvýšena výpočetní náročnost.



Obrázek 13: Schéma algoritmu VBM3D

4.3 Barevný obraz

Všechny výše uvedené metody jsou obvykle zaměřeny na práci s šedotónovým obrazem. U barevného obrazu by bylo možné aplikovat algoritmy pro potlačení šumu postupně na všechny barevné kanály RGB. Vzhledem k vysoké korelaci mezi jednotlivými kanály a faktu, že lidské oko není citlivé na vysoké prostorové kmitočty v barvě, by bylo vhodnější provést konverzi do jiného barevného prostoru, který je složen z jiných komponent než z jednotlivých barevných kanálů, jako je barevný prostor RGB nebo jeho doplňková varianta CMY. Obvykle se využívá prostorů složených z jednoho jasového kanálu a dvou kanálů nesoucích informaci o barvě, jako je například barevný prostor LAB nebo YCbCr [21].

YCbCr je často využívaným barevným prostorem řešícím danou problematiku [22]. Jedná se o barevný prostor s jedním jasovým kanálem Y a dvěma chrominančními Cr a Cb. Konverze do tohoto barevného prostoru ze standardního RGB prostoru není nikterak složitá. Jas je dán váhovaným součtem jednotlivých barevných kanálů, čímž obsahuje i vysoké prostorové kmitočty. Rozdílové složky Cb a Cr jsou vytvořeny rozdílem dané barevné složky a jasu – tedy B-Y a R-Y. Díky tomu v nich dochází k potlačení vyšších prostorových kmitočtů, a tedy i k výraznému potlačení šumu v daných kanálech. Převod z prostoru RGB do prostoru YCbCr může být popsán následujícími rovnicemi:

$$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} = \begin{bmatrix} 0 \\ 128 \\ 128 \end{bmatrix} + \begin{bmatrix} 0,299 & 0,587 & 0,114 \\ -0,169 & -0,331 & 0,500 \\ 0,500 & -0,419 & -0,081 \end{bmatrix} \cdot \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (4.4)$$

Samotný algoritmus pro potlačení šumu můžeme dále provádět pouze nad jasovým kanálem. To je jak z hlediska výpočetní náročnosti, tak skutečnosti, že se v práci dále budeme zabývat reálným bezpečnostním systémem, u kterého není kvalita barevného podání rozhodujícím faktorem, podstatně výhodnější.

5 Analyzovaný snímací systém

Pro další práci ověřování účinnosti algoritmů na potlačení šumu bylo vhodné analyzovat reálný snímací systém, ke zjištění potřebných parametrů šumu. Tyto parametry pak využívám v dalších kapitolách pro modelování šumu, kterým tento systém degraduje výsledný obraz.

Vybral jsem si funkční bezpečnostní kamerový systém, který sestával z dvojice kamer s možností infračerveného přisvícení pro snímání za velmi špatných světelných podmínek, typicky v noci. Jednalo se o exteriérové kamery značky HIKVISION. Konkrétně se jednalo o model DS-2CD2232-I5. Jedná se o nízkorozpočtovou bezpečnostní IP kameru pro širokou veřejnost. Podrobnější technické informace o použité kameře jsou uvedeny v následující podkapitole.



Obrázek 14: Kamera HIKVISION analyzovaného systému²

5.1 Parametry snímání

Výstup z bezpečnostních kamer byl zprostředkován formou webového rozhraní, které umožňovalo nastavení pouze některých parametrů, jako je omezení maximálního rozlišení, datového toku a snímkové frekvence. Bohužel nebylo možné většinu parametrů pevně nastavit, ale pouze omezit. Výslednou expozici si tak dle omezení řídila vnitřní elektronika kamery sama tak, aby bylo dosaženo optimálního výsledku vzhledem k nastaveným omezením. Kamera disponovala 4mm objektivem se světelností $f/2,0$. Snímacím prvkem byl čip CMOS o velikosti $1/3''$. Ekvivalentní ohnisková vzdálenost byla tedy 29 mm. Rychlost závěrky se mohla pohybovat v rozmezí od $1/3$ s do $1/100\,000$ s. Minimální osvětlení bez použití IR přísvitu se zapnutým AGC bylo $0,07$ lx, s IR přísvitem pak 0 lx dle katalogového listu výrobce.

Výstupním formátem k dalšímu zpracování byla bohužel pouze již komprimovaná videosekvence ve formátu MPEG-4. Konkrétně se jednalo o kodek AVC1, profilu Main 5, nastavením skupiny $GOP_{1,12}$ a bitovým tokem 4 Mbps. Bitová hloubka byla standardní – 8 bitů. Byl použit barevný model YUV s podvzorkováním barvonosných složek v poměru 4:2:0 v barevném prostoru ITU-R BT.709. Video bylo uloženo v kontejneru mp4 s 12 snímky za vteřinu v rozlišení 2048×1536 obrazových bodů.

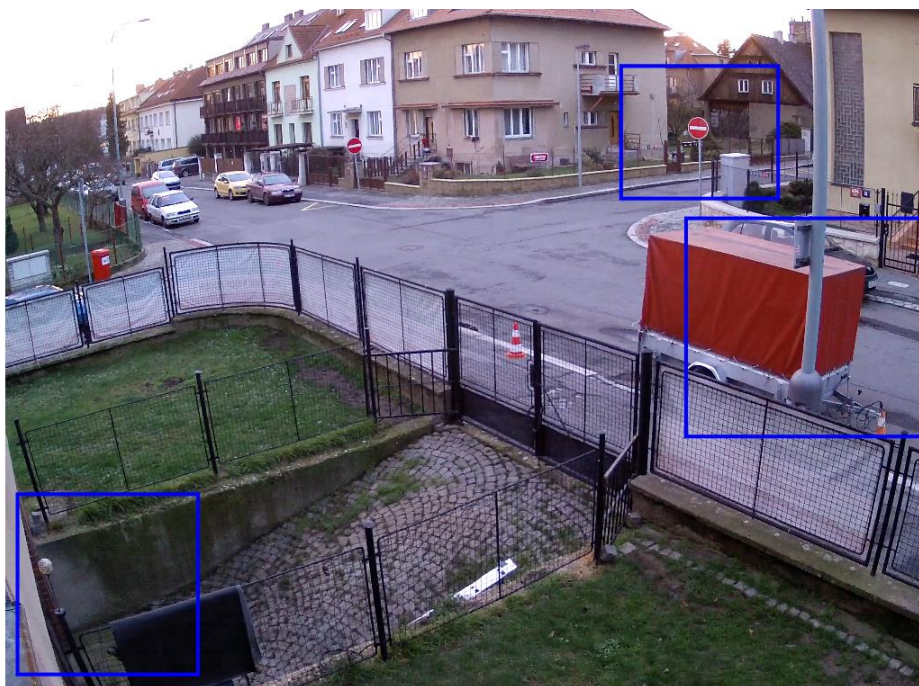
² Převzato z http://www.hikvision.com/us/Products_1_10533_i7698.html

5.2 Snímací podmínky

Jelikož se jednalo o venkovní bezpečnostní systém v nepřetržitém provozu, bylo obtížné nasimulovat některé snímací podmínky. Pro další zpracování byla proto využita reálná data nasnímaná během několika dnů běžného provozu s různými povětrnostními podmínkami (zataženo, polojasno, slunečno) a denním cyklem (den, večer, noc). Z takto vytvořené databáze byly dále vybrány časové úseky s optimálními (ideálně konstantními) světelnými podmínkami v dané kategorii a ty pak sloužily jako výchozí sekvence pro další analýzu.

5.3 Snímaná scéna

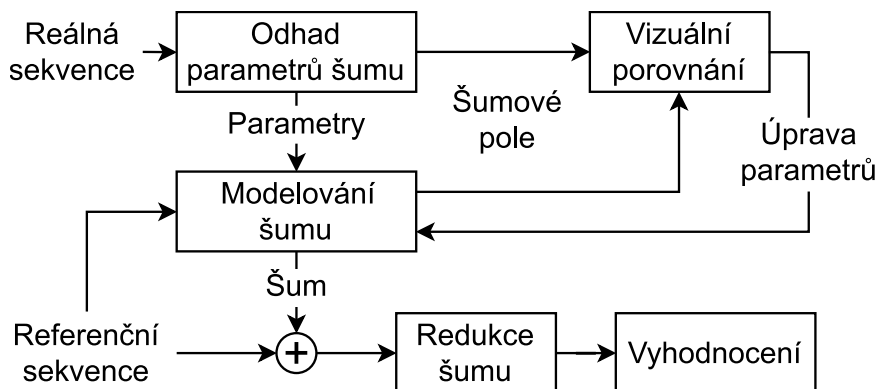
Snímanou scénou byl statický pohled na křižovatku v příměstské čtvrti. Jednalo se o celek centrováný do středu místní křižovatky dvou ulic z výšky zhruba čtyř metrů nad zemí s mírným nakloněním směrem k zemi. Vzdálenost k objektu ve středu obrazu byla zhruba dvacet metrů. Více než polovina snímku byla tvořena statickými objekty, převážně parkujícími automobily a pozadím, které tvořily okolní stavby. Pro další zpracování byly vybírány jednotlivé výřezy míst s minimem pohybu a různým obsahem pro určení šumových parametrů daného systému.



Obrázek 15: Snímaná scéna s vyznačenými výřezy pro další testy

6 Modelování

Dalším bodem zadání diplomové práce bylo ověření účinnosti vybraných algoritmů na redukcí šumu pomocí modelování. V následujících kapitolách je tedy popsána nezbytná analýza šumu, který vykazoval snímací systém, a postup, kterým byl daný systém modelován pro porovnání vybraných algoritmů pro potlačení šumu na základě objektivních metrik s plnou referencí. V následujícím blokovém schématu je znázorněn postup, který je pak podrobněji rozepsán v odpovídajících podkapitolách.

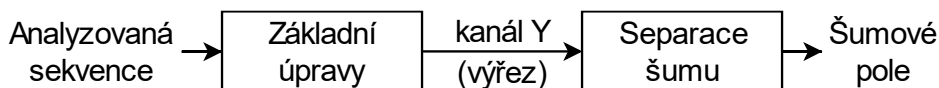


Obrázek 16: Schéma simulace reálného systému

Vytvořené funkce a skripty jsou uzpůsobeny tak, aby uživatel mohl jednoduše analyzovat šum u zvolené videosekvence a následně ověřovat účinnosti vybraných algoritmů pomocí jednotlivých objektivních metrik.

6.1 Odhad parametrů šumu

K odhadu parametrů šumu, který daný snímací systém vykazuje, byla vybrána sekvence s konstantními snímacími podmínkami. Dle uvedeného diagramu byla podrobena potřebným krokům, aby bylo možné odhadnout základní vlastnosti šumu.



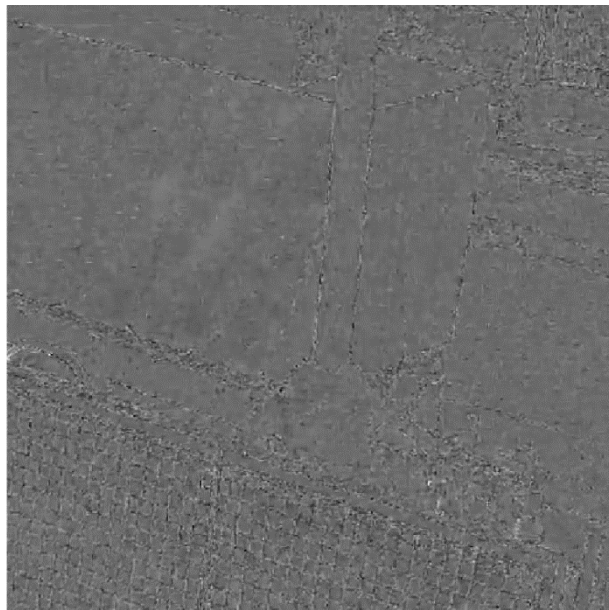
Obrázek 17: Schéma separace šumového pole

U vstupní analyzované sekvence byly provedeny základní úpravy, kde byl vybrán výřez míst obsahující různé objekty, především místa s kvazihomogenními plochami s různou intenzitou bez dalšího pohybu o zvolené délce – v našem případě se jednalo o 300 snímků. Sekvence byla následně převedena do barevného prostoru YCbCr. Jak již bylo zmíněno v předchozích kapitolách, tento barevný prostor redukuje v rozdílových chrominančních kanálech vysoké prostorové kmitočty – tedy i přidaný šum. Další kroky analýzy jsme díky tomu prováděli

především nad jasovým kanálem Y . O tyto úpravy se starají vytvořené funkce *crop.m* a *frames.m*. V dalším bloku byl ze všech snímků sekvence postupným průměrováním v časové oblasti vytvořen jeden snímek I_{avg} . Jednalo se o snímek se znatelně potlačeným šumem. Tímto snímkem jsme se pak jednoduchým odečítáním od jednotlivých snímků sekvence pokusili oddělit šum od odhadu původního signálu (za předpokladu, že se jedná o aditivní šum a snímek I_{avg} jsme považovali za odhad původního signálu neovlivněný šumem) a vytvořit tak sekvenci obsahující pouze přidaný šum – sekvenci šumového pole. Tento blok zajišťuje funkce *avg_diff.m*. Výpočet jednotlivých snímků šumového pole je znázorněn pomocí následujícího vztahu, kde $I(n)$ je původní, šumem degradovaný snímek sekvence v čase n

$$I_{nf}(n) = I(n) - I_{avg} . \quad (6.1)$$

Z šumového pole (Obrázek 18) je na první pohled zřejmé, že se nebude jednat o běžný aditivní šum, jelikož tím by v tomto případě byla pouze matice náhodných hodnot o rozměrech obrazu, avšak bez viditelných obrysů jednotlivých objektů. Šum je tedy buďto závislý na původním signálu, nebo snímek I_{avg} není snímek s ideálně potlačeným šumem.

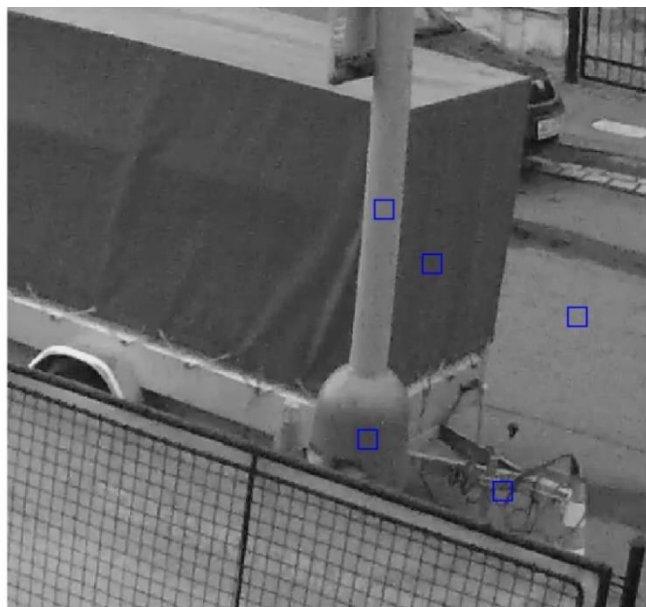


Obrázek 18: Šumové pole vybraného výřezu

Při bližším zkoumání šumového pole je ale jasné, že se nebude jednat ani o čistě signálově závislý šum. Zde totiž plochy s různou intenzitou pixelů vykazují obdobný šum, nikoli rozdílný v závislosti na původní intenzitě signálu, což je zásadní vlastnost, kterou se signálově závislý šum obvykle vyznačuje. Byla proto nutná další analýza sepsaná v další podkapitole.

6.1.1 Analýza signálové závislosti

Aby bylo možné ověřit signálovou závislost šumu, bylo vybráno několik menších výřezů o velikosti 15x15 obrazových bodů (Obrázek 19). V těchto místech, která obsahovala jednolitě plochy bez hran a přechodů, ovšem s jinou intenzitou (odstínem), byl spočítán rozptyl, střední hodnota a rozptyl podělený kvadrátem střední hodnoty – referenční rozptyl [23].



Obrázek 19: Výřez s vyznačenými vzorky pro analýzu šumových parametrů

$$rvar = \frac{1}{N-1} \frac{\sum_{i=1}^n |A_i - \mu|^2}{\mu^2} \quad (6.2)$$

Z uvedeného vztahu referenčního rozptylu, kde je vrchní část zlomku rozptyl, ve kterém A značí hodnotu pixelu, N je počet pixelů testovaného vzorku a μ je střední hodnotou, je patrné že se rozptyl hodnot v místech s vyšší intenzitou dělí větším číslem průměrné hodnoty. Tím pádem, pokud by se jednalo o multiplikativní, signálově závislý šum, hodnoty referenčního rozptylu by se pro jednotlivé vzorky s různou intenzitou příliš neměnily v porovnání se změnou normálního rozptylu. Oproti tomu, kdyby se jednalo o aditivní, signálově nezávislý šum, změny by byly přesně opačné. Normální rozptyl by zůstával podobný a hodnoty referenčního rozptylu by se znatelně měnily. Tyto výsledky nám následně ukázal výpočet, ze kterého můžeme usuzovat, že se jedná o formu aditivního šumu. Pro další ověření jsme provedli odečet průměrného snímku I_{avg} od sekvence kontaminované šumem po transformaci do logaritmu. Po této transformaci lze jednoduše odečíst multiplikativní šum. Očekávané šumové pole by po zpětné transformaci byla opět jen matice náhodných hodnot. To se nám ale ani tímto testem nepotvrdilo. Šumové pole vykazovalo opět jistou závislost na původním obrazu. Rozhodli jsme se tedy vyloučit možnost, že se jedná o běžný multiplikativní šum.

6.1.2 Rozdělení šumu

Pro určení rozdělení šumu jsme vykreslili histogramy jednotlivých zkoumaných vzorků s optimální šířkou binu určenou následujícím vztahem

$$bw = 2 \cdot (x_{0,75} - x_{0,25}) \cdot N^{-\frac{1}{3}}, \quad (6.3)$$

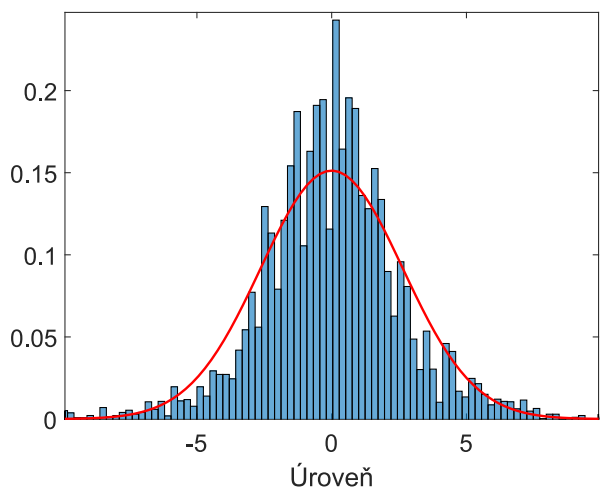
kde $x_{0,75} - x_{0,25}$ je mezikvartilové rozpětí a N je počet pixelů ve zkoumaném vzorku [24]. Následně jsme se pomocí aproximačního modelu funkce hustoty pravděpodobnosti snažili určit rozdělení šumu. Z grafů uvedených na další straně jsme usuzovali, že se jedná o šum s hodnotami blízcími se normálnímu rozdělení a nulovou střední hodnotou. Pro ověření normality jsme dále pro jeden vybraný snímek aplikovali Lillieforsův test normality, kdy nám u většiny vzorků byla potvrzena nulová hypotéza – tedy že se jedná o soubor dat s normálním rozdělením. Směrodatná odchylka šumu se v místech obsahujících jednodité plochy pohybovala mezi hodnotami 1 až 5 v závislosti na jednotlivých podmínkách snímání. V místech obsahujících hrany byla směrodatná odchylka šumu násobně větší. Obvykle se jednalo o hodnoty v rozmezí 4 až 11. Konkrétní hodnoty jsou uvedeny v následující tabulce. Hodnoty v nočním režimu s infračerveným přísvitem jsou podstatně nižší než za ostatních podmínek. Tento fakt je způsoben tím, že infračervený přísvit je poměrně silný a v místech, která kamera snímá, je instalováno pouliční osvětlení. Z těchto důvodů jsou záběry někdy až přexponované a některá místa jsou v saturaci. Tím pádem u nich nemůžeme pozorovat šum.

Tabulka 1: Směrodatné odchylky šumu analyzovaného systému

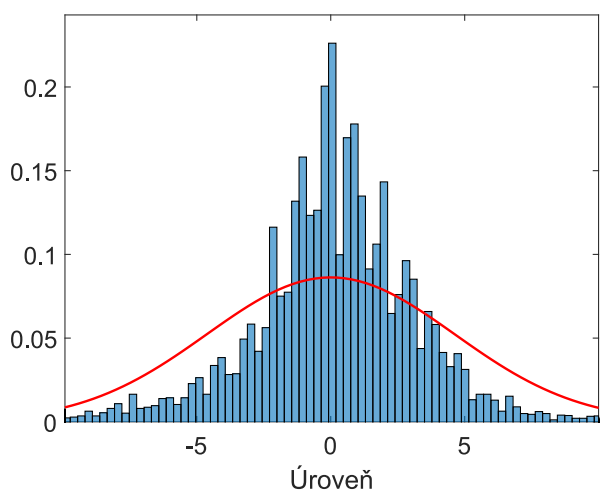
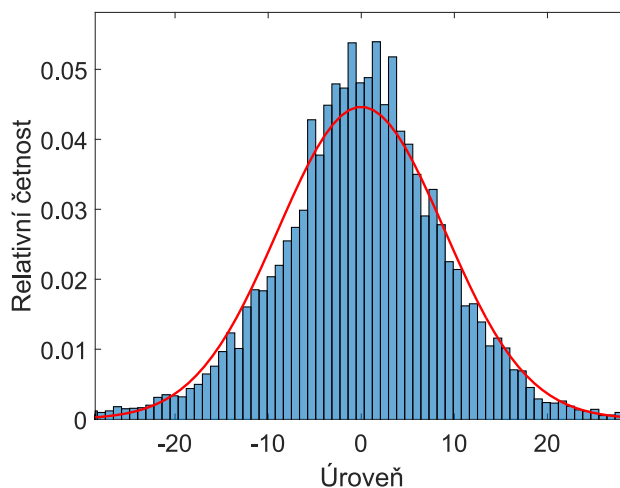
Světelné podmínky snímané scény	Průměrná směrodatná odchylka šumu ³	
	hladké plochy	místa obsahující hrany
špatné	2,67	8,89
velmi špatné	4,68	10,39
infračervené přísvětlení	1,52	4,59

Na další straně jsou uvedeny průměrné grafy histogramů z míst obsahujících jednodité plochy a míst obsahujících hrany v obraze. Červeně je znázorněna aproximace normálním rozdělením. O všechny výše uvedené výpočty, definování umístění, velikosti i počtu vzorků a následné vykreslení grafů se starají funkce *noise_loc.m* a *noise_type.m*.

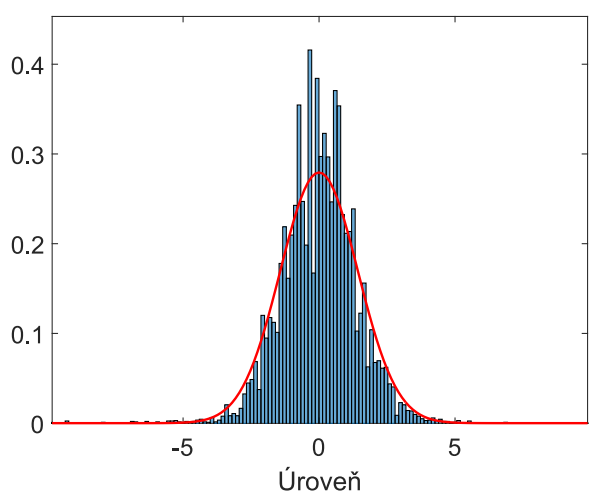
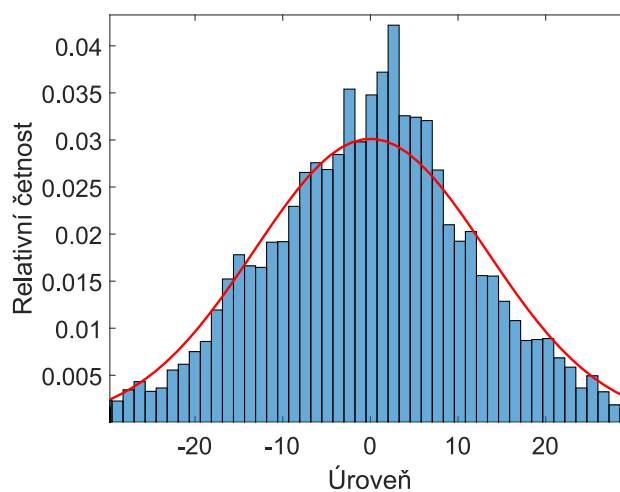
³ Průměrné hodnoty z měření ve 20 místech o velikosti 8x8 obrazových bodů



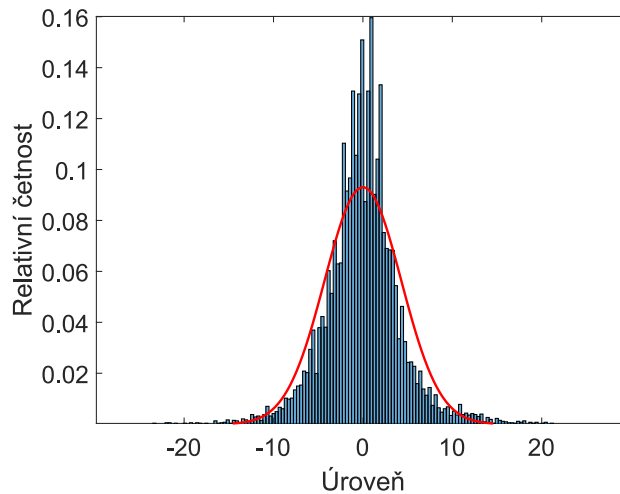
Obrázek 20: Průměrný histogram šumu za špatných podmínek (plochy/hrany)



Obrázek 21: Průměrný histogram šumu za velmi špatných podmínek (plochy/hrany)



Obrázek 22: Průměrný histogram šumu s IR přísvitem (plochy/hrany)



Dalším poznatkem při zkoumání šumového pole byla jeho jasně bloková struktura. Tento fakt společně se zesílením šumu v oblasti hran a jeho proměnlivou časovou změnou dokazuje, že výsledný šum je silně ovlivněn ztrátovým kompresním algoritmem výstupního videa. Tím pádem byl šum, jednak z důvodu hrubšího kvantování vyšších prostorových kmitočtů, tak i vnitřním algoritmem kamery pro potlačení šumu, částečně redukován v oblastech hladkých jednolitých ploch. Vnitřní algoritmus kamery pro potlačení šumu může být realizován nejrůznějšími filtry a metodami popsány v teoretické části. V našem případě to ale velmi ztěžuje následné modelování daného šumu, jelikož jsou jeho charakteristiky aplikací kompresních algoritmů silně ovlivněny, stejně jako původní obrazová data. Vzhledem k našemu zaměření na reálný systém, u kterého nemáme k dispozici surová data a nemůžeme příliš ovlivňovat snímací podmínky, jsme se rozhodli použít zjednodušení, jelikož přesné modelování šumu u daného reálného systému by bylo složitostí nad rámec této práce.

6.1.3 Barevné kanály

Dále jsme ověřili, že při snímání v nočním režimu (s infračerveným přísvitem) se kamera pravděpodobně přepne do šedotónového módu, jelikož byly jednotlivé RGB kanály totožné a po převodu do barevného prostoru YCbCr pak rozdílové chrominanční složky neobsahovaly žádnou informaci. V normálním režimu byl šum v jednotlivých kanálech velmi podobný, téměř identický. Tyto aspekty společně s již dříve zmíněnými nám umožnily barevnou informaci při ověřování účinnosti algoritmů na potlačení šumu zanedbat a zaměřit se především na jasový kanál.

6.2 Realizace testovacího modelu

Abychom mohli srovnávat algoritmy na potlačení šumu, na základě objektivních metrik, je nutné mít referenční videosekvenci neobsahující žádný šum. Do takové sekvence budeme pak uměle vytvořený šum přidávat, abychom získali model odpovídající reálnému analyzovanému systému. Jak jsem již zmínil v předchozí podkapitole, v našem případě budeme používat model, který se bude reálnému systému pouze přibližovat, jelikož je přesné modelování šumu příliš náročné. Jako kritérium podobnosti jsme zvolili vizuální porovnání vytvořeného a reálného šumového pole, jelikož se jedná o náhodnou veličinu, a tak nejsme schopni přesně určit, do jaké míry se přiblížíme šumu z reálného systému.

Je několik možností, jak referenční videosekvenci získat. První, pro náš případ ale ne příliš vhodnou metodou, je vytvoření referenční videosekvence pomocí průměrování reálných dat kontaminovaných šumem. Tato metoda se obecně nehodí pro videosekvence, jelikož obsahují pohyb. Je vhodná spíše pro série statických fotografií, což ale není naším cílem zkoumání. Další, již vhodnější metodou je použití reálného systému za ideálních světelných podmínek a tedy předpokladu, že taková sekvence bude vykazovat velmi dobrý odstup signálu od šumu. Výhodou této metody by bylo použití identické optické soustavy jako u analyzovaných dat při snímání za zhoršených podmínek. Tuto metodu ale můžeme používat pouze za předpokladu, že šum v referenčních sekvencích bude mít několikanásobně menší intenzitu než šum později přidaný při modelování, aby jej bylo možné zanedbat. Poslední možností je natočit sekvence kvalitní videokamerou za vhodných a kontrolovatelných podmínek snímání. U takové sekvence můžeme zajistit minimální úroveň šumu a neměnnou expozici v průběhu snímání, která by pochopitelně šum také dále ovlivňovala. A to je také metoda, kterou jsme se rozhodli pro naše modelování a další testování zvolit.

6.2.1 Tvorba referenční sekvence

Pro natočení referenční sekvence s minimálním obsahem šumu jsme využili digitální jednookou zrcadlovku Canon EOS 550D s objektivem Sigma 17-70mm f/2,8-4,0 DC Macro OS HSM. Jedná se o digitální zrcadlovku, která umožňuje snímat kvalitní videosekvence ve vysokém rozlišení, odpovídající našim potřebám. Ve spojení s dodatečným firmwarem Magic Lantern⁴ umožňuje nastavení potřebných snímacích parametrů jako u profesionální videokamery.

Technické informace o snímacím prvku fotoaparátu Canon 550D:

Typ snímače:	CMOS
Maximální rozlišení:	18 MPX
Velikost snímače:	APS-C (22,3 × 14,9 mm)

⁴ <http://www.magiclantern.fm/>

Snímací parametry jsou z hlediska výstupního formátu srovnatelné s reálným systémem. Jedná se o kodek AVC1, profilu Baseline 5, nastavením skupiny GOP_{1,15} s bitovou hloubkou 8 bitů v barevném modelu YUV s podvzorkováním barvosných složek v poměru 4:2:0 a barevném prostoru ITU-R BT.709. Pro lepší ověření účinnosti algoritmů pro potlačení šumu u sekvencí obsahujících pohyb byla snímková frekvence nastavena na 25 snímků za vteřinu a rozlišení videa bylo standardních 1920×1080 obrazových bodů.

Snímaná scéna byla uzpůsobena tak, aby odpovídala scéně snímané analyzovaným systémem. Jednalo se tedy o celek v exteriéru zabírající ulici z místa odpovídajícího reálnému systému. Sekvence byla pro jednotlivé snímací podmínky – intenzitu a druh šumu natočena vždy pouze jedna, aby byly zajištěny stejné výchozí podmínky pro srovnávání účinnosti jednotlivých algoritmů. Pro zkoumání účinnosti redukce šumu u pohybujících se objektů bylo natočeno několik scén s různým druhem pohybu odpovídajícím reálným podmínkám. V následující tabulce jsou uvedeny nasnímané referenční sekvence a jejich základní popis.

Tabulka 2: Seznam referenčních sekvencí

Označení sekvence	Stručný popis	Typ pohybu
1	Projíždějící vozidlo	Rychlý
2	Parkující vozidlo	Pomalý
3	Procházející chodci	Pomalý

Aby byly zajištěny stejné výchozí parametry pro všechny druhy pohybu, byly sekvence natočeny s pevnou expozicí.

Tabulka 3: Parametry referenčních sekvencí

Ohnisková vzdálenost	Clonové číslo	Doba expozice	Citlivost ISO	Barevná teplota
28 mm ⁵	f/8,0	1/50 s	100	5600 K

Na další stránce je ukázka snímané referenční scény (Obrázek 23), ze které byly pro další práci používány jednotlivé výřezy s oblastmi, které se zaměřovaly na daný druh pohybu. Tyto oblasti byly voleny také proto, aby se snížila výsledná výpočetní náročnost.

⁵ Ekvivalentní ohnisková vzdálenost je zde 45 mm

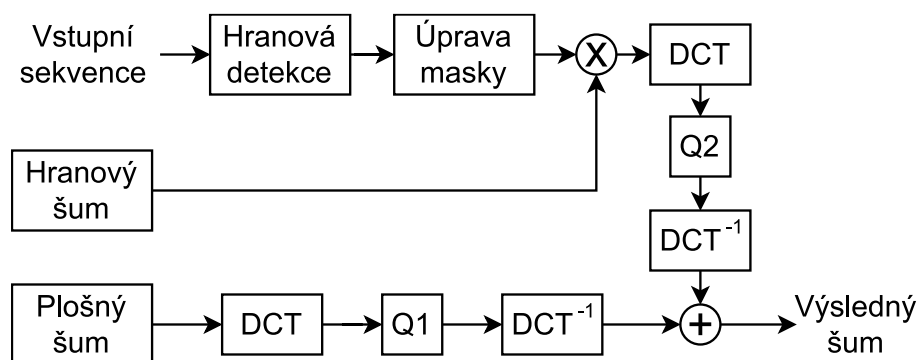


Obrázek 23: Ukázka nasnímané referenční sekvence

6.2.2 Modelování šumu

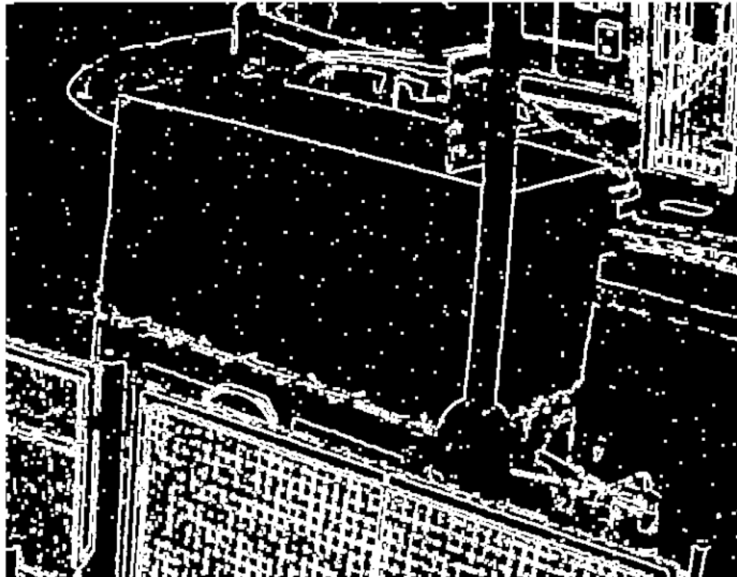
Pro základní testování účinnosti algoritmů pro potlačení šumu jsme se rozhodli využít běžně používanou aproximaci s parametry, které jsme získali analýzou šumu u reálného systému. Tedy že se bude jednat o aditivní bílý Gaussovský šum s nulovou střední hodnotou a se směrodatnou odchylkou získanou průměrem z analyzovaných dat. Tento model se používá, jak jsem již v úvodních kapitolách zmiňoval, jako nejběžnější aproximační model. Hodnota směrodatné odchylky šumu bude v závislosti na poměru obsahu hran a jednodolých ploch v obrázku odpovídajícím způsobem pro dané simulované snímací podmínky zvolena.

Abychom se modelem ale více přiblížili analyzovanému systému, který vykazoval šum silně ovlivněný kompresními algoritmy, vytvořili jsme ještě jeden nekonvenční model šumu. Postup, kterým jsme takové šumové pole realizovali, je ilustrován na následujícím schématu.



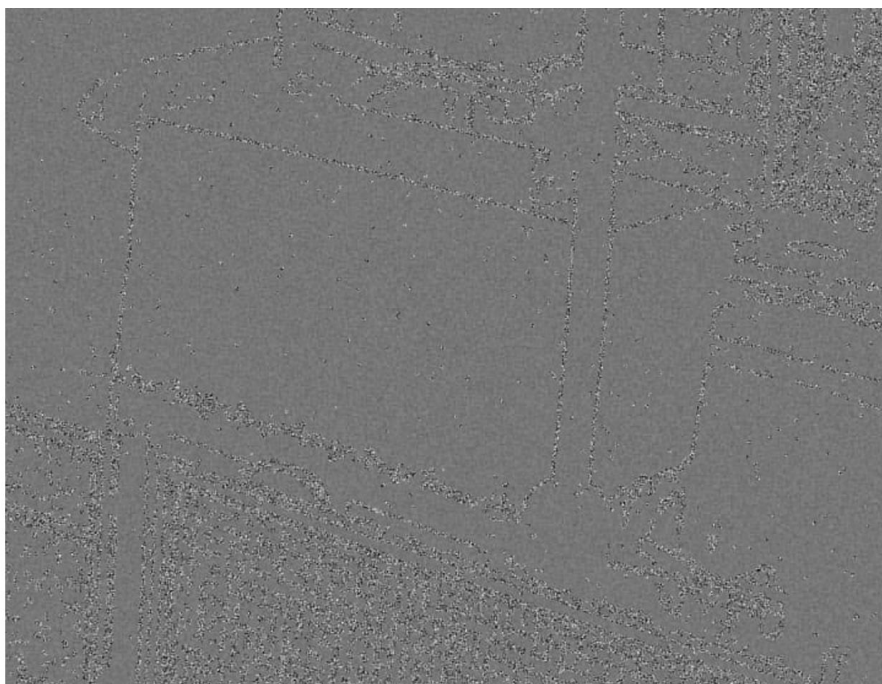
Obrázek 24: Schéma tvorby šumu

Nejprve jsme vytvořili dva druhy šumu s normálním rozdělením a nulovou střední hodnotou. Pro modelování výraznějšího šumu v místech obsahujících hrany v obraze (hranový šum) jsme zvolili větší rozptyl dle analýzy šumu reálného systému. Pomocí jednoduché hranové detekce jsme pro každý snímek vstupní referenční sekvence detekovali hrany v obraze. K tomu jsme využili interní funkce pro rozpoznání hran se sobelovou konvoluční maskou. Vzniklé hrany jsme dále zesílili a vzniklou masku zatížili impulsním šumem typu sůl a pepř, jelikož obdobné jevy vykazovalo i reálné šumové pole.

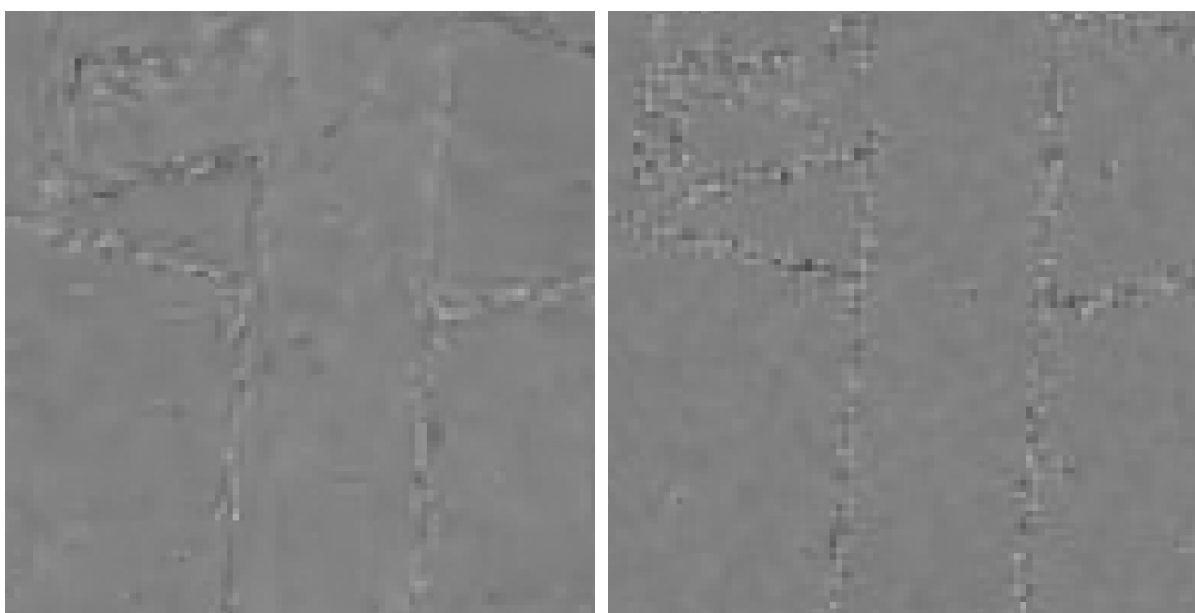


Obrázek 25: Maska pro zesílení šumu v oblastech hran

Hranový šum jsme pak vynásobili se vzniklou maskou (obrázek výše) a tím jsme vytvořili základ pro výraznější šum na hranách, který jsme později kombinovali s plošným šumem s menším rozptylem (plošný šum). Ještě před touto kombinací jsme oba druhy šumu komprimovali, abychom výslednému šumu dodali blokovou strukturu, kterou vykazoval analyzovaný systém, a potlačili tím typicky jemné zrnění. Kompresi jsme provedli pomocí dvourozměrné diskrétní kosinové transformace (pro rozklad jsme zvolili velikost 8x8 obrazových bodů) s různou kvantizační maticí pro plošný šum (Q1) a šum v oblastech hran (Q2). Na následujícím obrázku můžeme vidět vytvořené šumové pole. Modelované šumové pole při větším přiblížení vykazuje jistou podobnost s jeho reálnou předlohou. Šlo nám především o zvýraznění šumu v oblastech hran, což se nám touto metodou podařilo dosáhnout. Při ověřování účinnosti algoritmů pro potlačení šumu pak bude tento model o něco sofistikovanější, jelikož bude obsahovat šum s různým rozptylem a jistou signálovou závislostí.



Obrázek 26: Generované šumové pole (pro lepší viditelnost s roztaženým histogramem)



Obrázek 27: Detailní porovnání šumového pole (reálné/modelované)

Tvorbu šumu zajišťuje funkce *noise_gen.m*, která jako vstup vyžaduje referenční sekvenci, volbu modelu šumu (jednoduchý AWGN nebo výše uvedený nekonvenční model) a směrodatnou odchylku, resp. odchylky šumu v oblastech ploch a hran. Výstupem funkce je šumové pole, kterým se následně degraduje referenční sekvence.

6.3 Testované algoritmy

Pro testování účinnosti jsme vybrali několik algoritmů popisovaných v teoretické části od základních filtrací obrazu až po nejmodernější algoritmy pro časoprostorovou redukci šumu u videosekvencí. Jako úplně základní jsme zvolili ukázkou prostého rozostření pomocí průměrujícího Gaussova filtru popisovaného na začátku teoretické části této práce. Dále jsme chtěli demonstrovat vliv nebo spíše jakou má účinnost na daný typ šumu mediánový filtr. To jsou velmi jednoduché a základní operace s obrazem, které ukazujeme a porovnáváme spíše pro ukázkou. Jelikož se jedná o metody, resp. filtrace obrazu, které neberou v úvahu časovou změnu, budou postupně aplikovány na jednotlivé snímky sekvence nezávisle na sobě. Oba výše zmíněné filtry jsou implementovány jako vnitřní funkce MATLABu. Další níže popisované metody už nejsou jeho běžnou součástí, a proto jim budeme vždy věnovat krátkou podkapitolu.

6.3.1 Bilaterální filtr

Již pokročilejší a účinnější metodou na potlačení šumu v obraze, kterou budeme podrobovat testu, je bilaterální filtrace. Princip bilaterálního filtru je uveden v teoretické části práce (kapitola 2.2.2), takže zde již zmíním jen konkrétní specifika použité funkce. Jelikož se nejedná o funkce implementované v prostředí MATLAB, vybrali jsme si dostupnou implementaci filtru s názvem **Bilateral Filtering**⁶ od **Douglase Lanmana**, kde jsme využili soubor *bfilter2.m* a upravili jej pro naše potřeby. Parametry filtru jsme upravili tak, aby poskytovali co nejlepší možný výsledek z hlediska subjektivního porovnání výsledného obrazu. Nejedná se o metodu, která by využívala časové souvislosti snímků, a bude proto aplikována na jednotlivé snímky sekvence. Vstupními parametry funkce jsou degradovaný snímek, velikost okna okolo referenčního pixelu a hodnoty sigma ovlivňující míru zachování hran a vyhlazení obrazu.

6.3.2 Vlnková transformace

Jak jsem popisoval v teoretické části, velmi účinnou metodou pro redukci šumu jsou také algoritmy na bázi vlnkové transformace s vhodným prahováním detailních vlnkových koeficientů. Jednu takovou metodu jsme se také rozhodli zahrnout do našeho testování. Vybrali jsme dostupnou implementaci **NeighShrinkSURE denoising**⁷, kterou pro MATLAB vytvořil **Dengwen Zhou**. To je jeden z autorů metody, která využívá metody prahování SURE (kapitola 3.2.2) a blízkého okolí zvolené velikosti k optimalizaci prahování daného vlnkového koeficientu. Podrobnější informace o daném algoritmu jsou v literatuře [25]. Algoritmus jsme aplikovali nezávisle na každý snímek sekvence degradované šumem. Vstupními parametry

⁶ Dostupné z: <https://www.mathworks.com/matlabcentral/fileexchange/12191-bilateral-filtering>
[cit. 2018-01-06]

⁷ Dostupné z: <https://www.mathworks.com/matlabcentral/fileexchange/20705-neighshrinksure-denoising>
[cit. 2018-01-06]

funkce *NeighShrinkSUREdenoise.m* byl obrázek kontaminovaný šumem, směrodatná odchylka šumu, mateřská vlnka a stupeň dekompozice.

6.3.3 Kálmánův filtr

Obecná Kálmánova filtrace je uvedena v kapitole 4.1.2. Konkrétní algoritmus je zaměřen především na potlačení šumu u statických videosekvencí. Využívá časového okna, kdy jednotlivé hodnoty pixelů odhaduje z předchozího snímku a při každém dalším snímku odhad upravuje a zpřesňuje. Implementace je dostupná pod názvem **Kalman filter for noisy movies**⁸ od **Roba Campbella**. Na obsluhu je velmi jednoduchá, funkce s názvem *Kalman_Stack_Filter.m* vyžaduje na vstupu pouze trojrozměrnou matici obsahující sérii jednotlivých snímků videa. Dalšími volitelnými vstupy jsou úvodní odhad rozptylu šumu a síla filtru. Úvodní odhad rozptylu šumu nemá příliš velký vliv na výsledek. Síla filtru ovlivňuje, která hodnota bude mít na následující odhad větší vliv, jestli predikovaná, nebo pozorovaná. Podrobný popis konkrétní implementace je uveden v jejím popisu⁸. Výstupem algoritmu je pak matice filtrovaného videa o stejné dimenzi.

6.3.4 BM3D a VBM3D

Jako jednu z nejmodernějších a nejpokročilejších metod pro redukci šumu jsme si vybrali metody **BM3D** a **VBM3D**. Obě metody jsou pro vzdělávací a vědecké účely volně dostupné ze stránek Tampere University of Technology⁹. Jako první budeme testovat algoritmus z hlediska kvality a výpočetní náročnosti pro jednotlivé snímky bez vzájemné souvislosti. Tím bude dobře srovnatelná účinnost BM3D algoritmu s podstatně jednoduššími typy filtrů. Dále pak otestujeme variantu upravenou přímo pro videosekvence (VBM3D), která bere v úvahu i časovou souvislost jednotlivých snímků. Vstupem jednotlivých metod je obrázek kontaminovaný šumem, respektive videosekvence, v podobě buďto matic nebo přímo souborů v počítači. Dále pak volitelně referenční snímek, resp. videosekvence, pro výpočet objektivní metriky PSNR, směrodatná odchylka šumu, pokud je známa, a možnost aktivace výpisu informací a uložení výsledného snímku do souboru.

⁸ Dostupné z: <https://www.mathworks.com/matlabcentral/fileexchange/26334-kalman-filter-for-noisy-movies> [cit. 2018-01-06]

⁹ Dostupné z: http://www.cs.tut.fi/~foi/GCF-BM3D/index.html#ref_software [cit. 2018-01-06]

6.4 Metriky hodnocení

Jedním z hlavních cílů této práce bylo objektivní ověření účinnosti vybraných algoritmů pro potlačení šumu a jejich vzájemné srovnání. Vzhledem k tomu, že reálný systém simulujeme, tak máme k dispozici jak sekvenci degradovanou šumem, tak její čistý originál – referenci. Bylo proto vhodné zvolit hodnocení účinnosti algoritmů pro potlačení šumu na základě objektivních metrik využívajících právě plnou referenci. Zvolili jsme dvě základní, velmi často používané a dvě pokročilejší beroucí v úvahu lidský vizuální systém. Existuje řada objektivních metrik, my se ale budeme zabývat těmi nejpoužívanějšími, konkrétně se bude jednat o metriky MSE, PSNR, SSIM a VIF [26].

6.4.1 MSE

Asi nejjednodušší metrikou je základní měření MSE (Mean Square Error) [26] vyjadřující střední kvadratickou chybu mezi porovnávanými snímky. Jak můžeme vidět z následujícího vztahu, její výpočet je vcelku snadný a výpočetně nenáročný

$$MSE = \frac{1}{M} \frac{1}{N} \sum_{i=1}^M \sum_{j=1}^N (\hat{X}(i,j) - X(i,j))^2, \quad (6.4)$$

kde M a N je horizontální, resp. vertikální počet bodů vstupního obrazu, \hat{X} je testovaný snímek a X referenční snímek. Je zřejmé, že s rostoucí hodnotou MSE se obraz více odlišuje. Pokud bychom testovali dva identické snímky, hodnota by byla rovna 0 – tedy žádný rozdíl. Bohužel je tato metrika ovlivněna například poklesem jasu obrázku a nebyly při jejím návrhu uvažovány vlastnosti lidského vizuálního systému. Hodnota MSE je bezrozměrná.

6.4.2 PSNR

Další metrikou je PSNR (Peak Signal-to-Noise Ratio), tedy poměr špičkového signálu k šumu [26]. Tato metrika se velmi často objevuje ve většině vědeckých prací pro objektivní hodnocení kvality obrazu. Umožňuje srovnání snímků s rozdílnou bitovou hloubkou, jelikož je v ní tento parametr zahrnut jako maximální hodnota (peak). Vychází z výše popisované MSE a je dána následujícím vztahem

$$PSNR = 10 \log_{10} \left(\frac{MAX^2}{MSE} \right), \quad (6.5)$$

kde MAX je maximální hodnotou obrazového bodu. U běžné 8bitové reprezentace obrazu se jedná o hodnotu 255. S rostoucí hodnotou PSNR se jedná o lepší výsledek, pokud bychom

testovali dva identické obrazy, výsledná hodnota by byla nekonečno. Z důvodu velkého dynamického rozsahu je uváděna v decibelové míře.

6.4.3 SSIM

Metrika SSIM (Structural Similarity index) oproti dvěma výše zmíněným bere v úvahu i psychovizuální model lidského vnímání. Měla by tedy lépe určit podobnost jednotlivých snímků tak, jak by je vyhodnotil pozorovatel. Je podstatně složitější než již zmíněné metriky a skládá se ze tří dílčích částí – jasového, kontrastního a strukturálního porovnání jednotlivých snímků. V jednotlivých obrazech vyhodnotí zvlášť daná specifika, která zkombinuje, čímž získáme index podobnosti, který může nabývat hodnot od -1 do 1, kdy hodnota 1 označuje shodné obrazy. Podrobnější informace o této metrice je možné nalézt v původní práci [27]. Metrika je implementována v prostředí MATLAB a pro ukázkou níže uvádím její výsledný vztah

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}. \quad (6.6)$$

6.4.4 VIF

Poslední metrikou, kterou budeme používat, je VIF (Visual Information Fidelity). Tato metrika je jedna z nejpřesnějších metrik z hlediska toho, jak by výsledek hodnotil pozorovatel. Výpočetně se jedná o nejsložitější metriku z námi používaných, která není interně implementována v MATLABu, a proto jsme sehnali její dostupnou implementaci¹⁰. Jádro metriky je ve vyhodnocení vzájemné informace, kterou obsahují porovnávané obrazy a jakou část z nich je schopen člověk vnímat. K tomu používá model lidského vnímání vytvořený pomocí Gaussovských směsí a vlnkové transformace, kdy v jednotlivých subpásmech zvlášť vypočítává dané informace. Výsledná hodnota je pak dána poměrem těchto informací ze zkresleného a referenčního snímku. Podrobnější popis dané problematiky je uveden v literatuře [28].

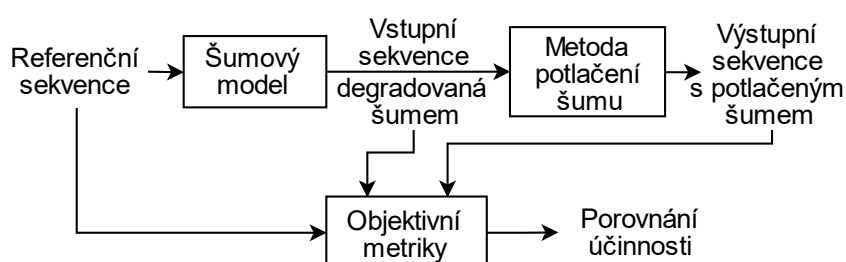
¹⁰ Dostupné z:
https://github.com/sattarab/image-quality-tools/tree/master/metrix_mux/metrix/vif/vifvec_release
[cit. 2018-01-06]

7 Vyhodnocení účinnosti algoritmů

Jedním z hlavních cílů této práce bylo srovnat účinnosti vybraných algoritmů pro redukci šumu na základě modelu pomocí několika objektivních metrik. Rozhodli jsme se porovnat jednak algoritmy zaměřené na videosekvence, tak jejich jednodušší varianty, které neberou v potaz časovou závislost jednotlivých snímků videosekvence. V následujících podkapitolách budeme postupně srovnávat jak účinnosti jednotlivých algoritmů z hlediska míry redukce šumu, tak z hlediska jejich výpočetní náročnosti. Toto porovnání budeme provádět na třech videosekvencích obsahujících různý druh pohybu a také pro dva modely přidaného šumu, které jsem popisoval v předchozích kapitolách.

7.1 Výsledky objektivního hodnocení redukce šumu

Vyhodnocení účinnosti algoritmů pro potlačení šumu probíhalo dle následujícího schématu, kdy jsme vždy porovnávali referenční sekvenci se vstupní sekvencí degradovanou šumem a referenční sekvenci s výstupní sekvencí s potlačeným šumem. Měření probíhalo pro každý snímek sekvence zvlášť. Do grafů jsme vynesli časový vývoj jednotlivých metrik a dále také průměrnou hodnotu, kterou budeme brát jako výsledek.

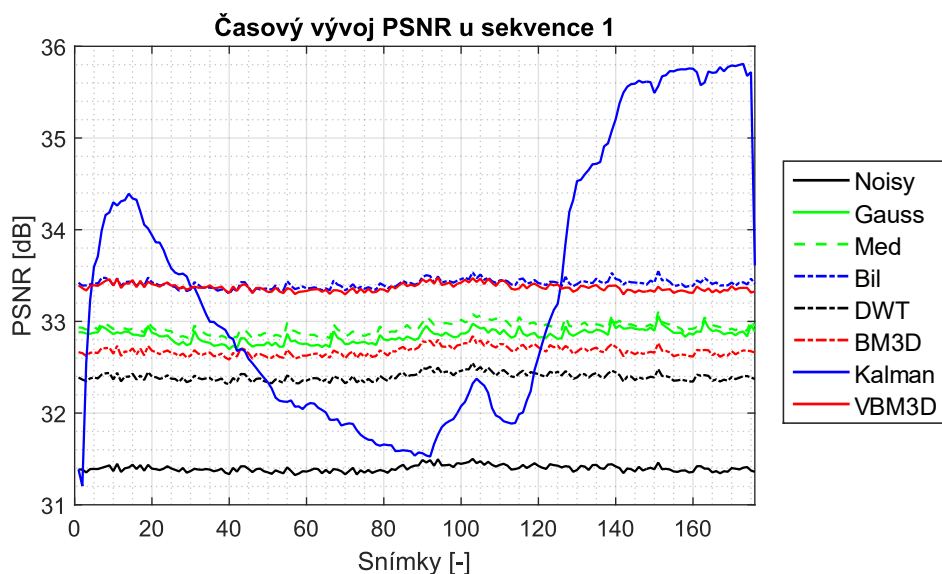


Obrázek 28: Schéma objektivního hodnocení

Dále v práci budu používat následující označení pro jednotlivé algoritmy uvedené v kapitolách 6.3 až 6.3.4:

Noisy	–	Sekvence kontaminovaná šumem
Gauss	–	Gaussův filtr
Med	–	Mediánový filtr
Bil	–	Bilaterální filtr
DWT	–	NeigShrinkSURE
BM3D	–	Block Matching 3D Algorithm pro jednotlivé snímky
Kalman	–	Kálmánův filtr
VBM3D	–	Block Matching 3D Algorithm pro videosekvence

První testovanou sekvencí je sekvence označená číslem 1 (viz Tabulka 2) s rychlým pohybem degradovaná nekonvenčním šumem, který jsme modelovali dle reálného systému (kapitola 6.2.2). Hodnoty směrodatné odchylky šumu byly nastaveny dle analyzovaných z reálného systému. Konkrétně se jednalo o hodnotu 2,7 pro šum v plochách a 9 pro šum v oblastech hran. Algoritmy, do kterých bylo možné jako vstupní parametr zadat i právě hodnotu směrodatné odchylky šumu byla zadána hodnota 5 aby byl zajištěn stejný odhad šumu, jelikož by pak některé algoritmy mohli lépe tento parametr odhadnout a tím by se pak zkreslilo výsledné porovnání. Tato hodnota byla zvolena vzhledem k poměru ploch a hran v obraze.



Obrázek 29: Graf vývoje PSNR u sekvence 1

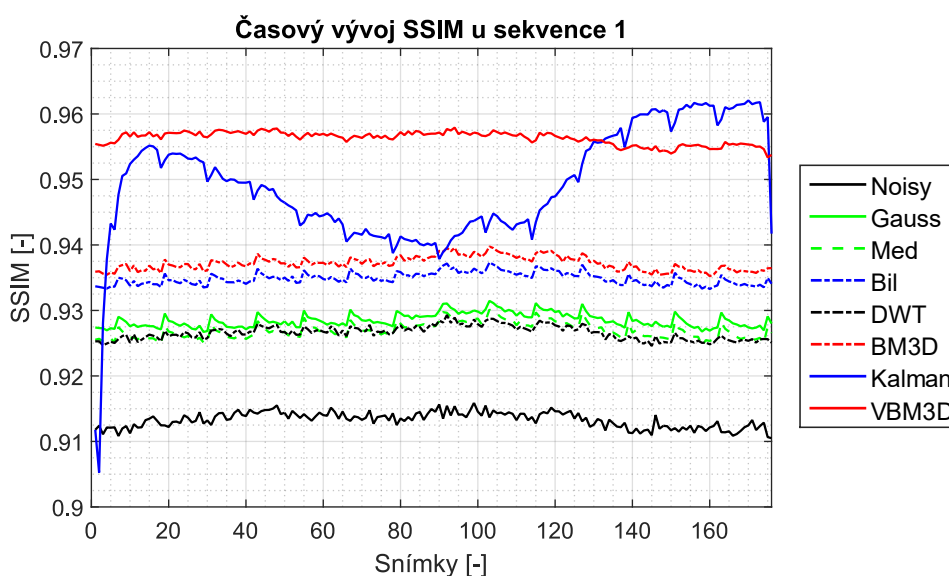


Obrázek 30: Ukázkové snímky sekvence číslo 1 (1, 30, 60, 90, 120 a 140)

Výše uvedený graf (Obrázek 29) ukazuje časový vývoj PSNR, kde jsou na první pohled jasně patrné zásadní rozdíly mezi jednotlivými algoritmy na potlačení šumu. Je zřejmé, že Kálmánův filtr je silně závislý na pohybu, respektive změně ve scéně. Především když se jedná o rychle se pohybující objekt. Ke konci sekvence Kálmánův filtr sice zvyšuje svou účinnost redukce

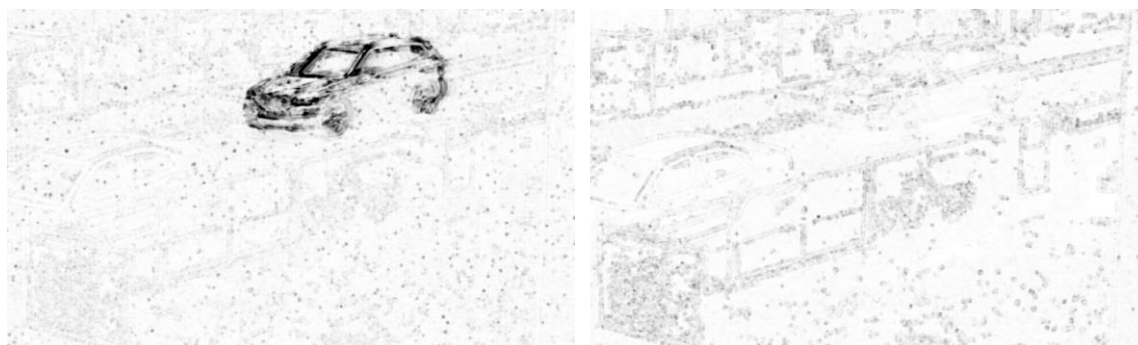
šumu oproti ostatním algoritmům, ale to je způsobeno zmizením pohybujícího se objektu, v našem případě jedoucího vozidla, které odjelo ze záběru.

Grafy MSE zde nebudu uvádět, jelikož z nich vycházejí hodnoty PSNR, které jsou uváděny ve většině vědeckých prací pro snadné srovnání. Zobrazit všechny grafy je možné z datové přílohy (viz Přílohy), kde jsou uloženy veškeré napočítané metriky. Hlavní vytvořený skript obsahuje část, která jednotlivé grafy umožňuje po načtení odpovídajících dat ve stejné podobě vykreslit. Jako další hodnotící metriky proto uvádím SSIM a VIF, které se také často objevují a již by měly být přesnější z hlediska toho, jak by sekvenci hodnotil reálný pozorovatel. Níže jsou uvedeny odpovídající časové vývoje.

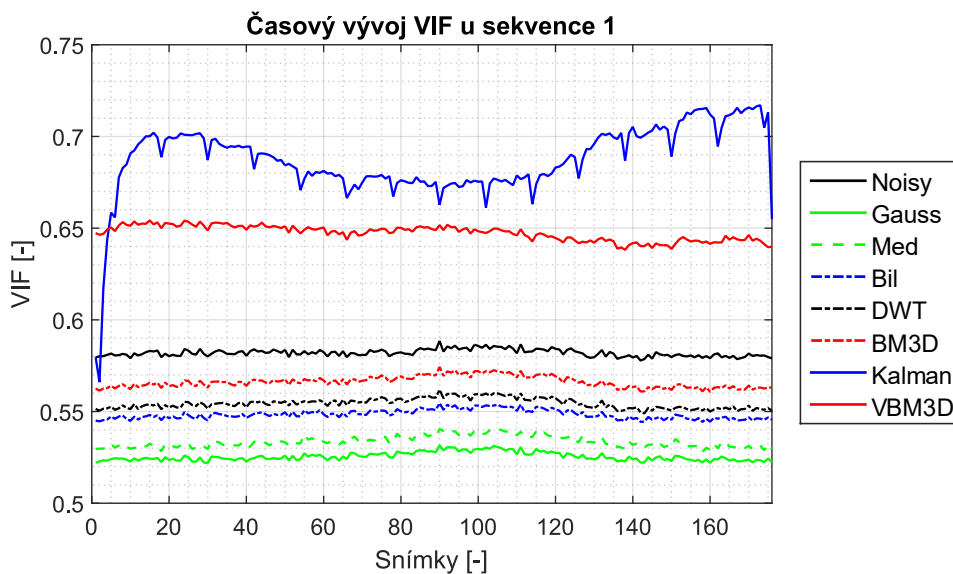


Obrázek 31: Graf vývoje SSIM u sekvence 1

Zde je již patrný rozdíl oproti PSNR. A to že z algoritmů zaměřených pouze na jednotlivé snímky je nejlepší algoritmus BM3D. Dle PSNR byla ale lépe hodnocena například i Gaussova filtrace, přestože byla na první pohled silně rozostřená. Níže můžeme vidět vývoj SSIM po ploše snímku. Konkrétně se jedná o Kálmánovu filtraci a VBM3D algoritmus v 75. snímku. Je vidět, že VBM3D o něco méně potlačuje silnější šum v oblastech hran (což může být způsobeno zadaným rozptylem, který byl zprůměrovaný dle poměru ploch a hran ve snímku), ale pohyb a impulsní šum potlačuje oproti algoritmu Kálmánovy filtrace naprosto bez problému.



Obrázek 32: Vývoj SSIM po ploše (tmavší je horší)



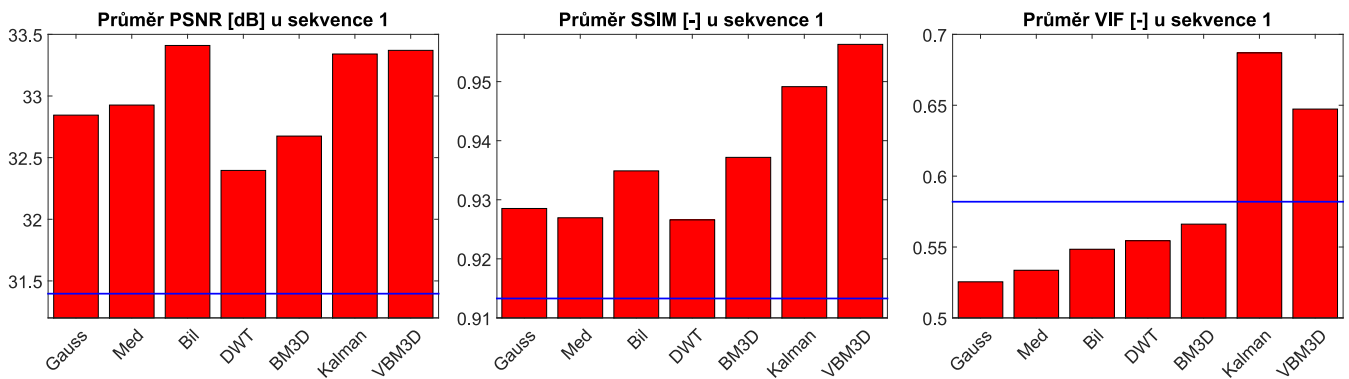
Obrázek 33: Graf vývoje VIF u sekvence 1

Z grafu časového vývoje VIF na předchozí straně vyplývají výsledky redukce šumu, které jsme očekávali. Tedy alespoň z hlediska pořadí jednotlivých algoritmů, kdy jsou seřazeny od nejhorších po nejlepší. A to i přesto, že jsou všechny algoritmy, zaměřené na jednotlivé snímky, hodnoceny hůře nežli sekvence degradovaná šumem. To je způsobeno tím, že v místech, kde sice viditelně potlačují šum, dochází ale také k určité ztrátě detailů. Například vlivem rozostření. A naopak v oblastech hran, kde je šum v tomto modelu ještě zesílen, se snaží detaily zachovat. Tím pádem zde šum také méně redukuje a ve výsledku jsou hodnoceny hůře než sekvence degradovaná šumem. Kálmánova filtrace je zde hodnocena nejlépe, jelikož poměrně silně potlačuje šum v místech bez pohybu, aniž by zde docházelo k jakékoli viditelné ztrátě detailů. Jelikož je velká část scény statická, tak je tento algoritmus hodnocen jako nejlepší i přesto, že zanechává poměrně výrazný obrys za pohybujícím se objektem, který je vidět na následujícím obrázku. Tento nežádoucí jev bývá označován jako ghosting.



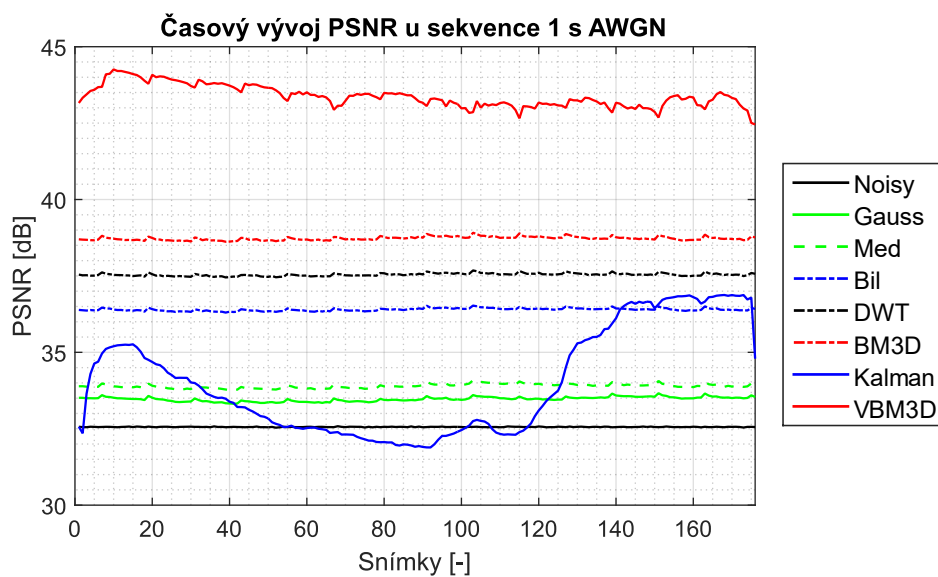
Obrázek 34: Ghosting u Kálmánovy filtrace

Na níže uvedených grafech jsou zobrazeny průměrné hodnoty jednotlivých metrik u dané metody pro potlačení šumu. Modře je znázorněna průměrná hodnota sekvence degradované šumem.



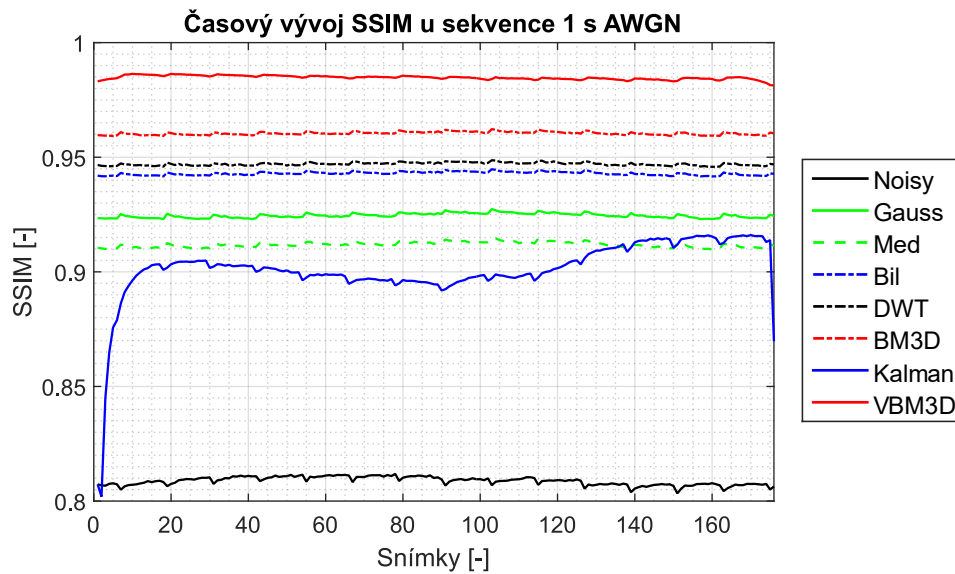
Obrázek 35: Grafy průměrných hodnot PSNR, SSIM a VIF u sekvence 1

Dále jsme podrobili testu stejnou sekvencí, kterou jsme ale kontaminovali jednodušším modelem s AWGN. Zde jsme nastavili směrodatnou odchylku šumu na hodnotu 6. Stejnou hodnotu jsme pak použili jako vstupní argumenty u algoritmů, které s touto hodnotou pracují, což by se mělo v tomto testu pozitivně projevit.



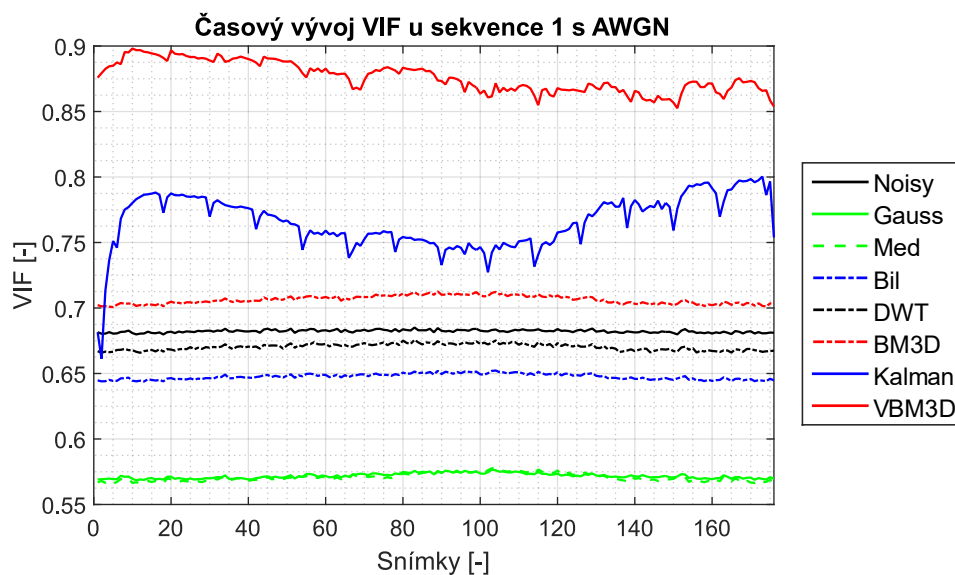
Obrázek 36: Graf vývoje PSNR u sekvence 1 s AWGN

Zde je patrný výrazný rozdíl v účinnosti jednotlivých algoritmů ve srovnání se sekvencí, kterou jsme kontaminovali vytvořeným nekonvenčním modelem šumu.



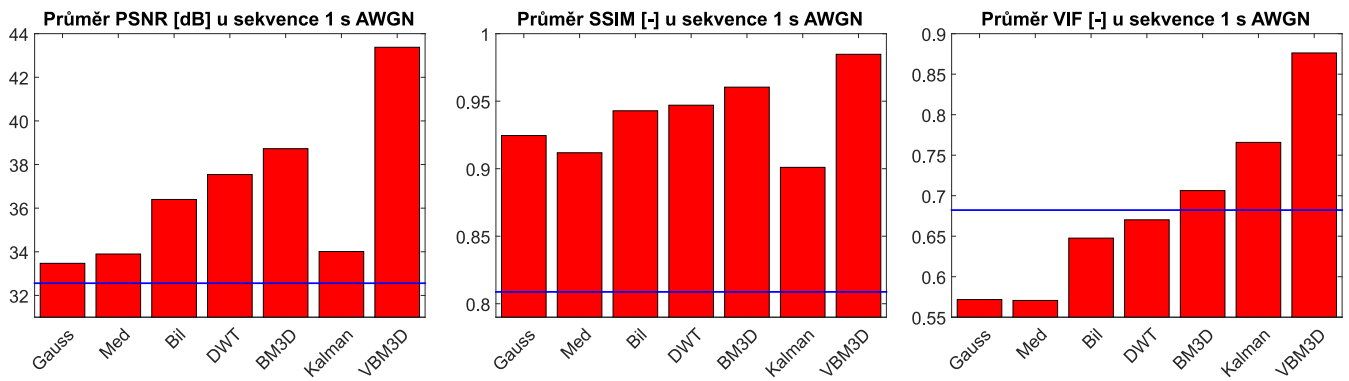
Obrázek 37: Graf vývoje SSIM u sekvence 1 s AWGN

Graf vývoje SSIM již podstatně více odpovídá subjektivnímu dojmu, kdy sekvence s potlačeným šumem pomocí algoritmu s Kálmánovou filtrací je pro člověka nejvíce rozdílná. A to především u pohybuujícího se objektu, který je silně rozostřen.



Obrázek 38: Graf vývoje VIF u sekvence 1 s AWGN

U metriky VIF je zajímavé, že i přesto, že došlo k potlačení intenzity šumu, jsou jednodušší algoritmy hodnoceny jako více rozdílné než sekvence degradovaná šumem. To je pravděpodobně způsobeno rozostřením některých částí snímku, a tak ztrátě struktur některých povrchů.

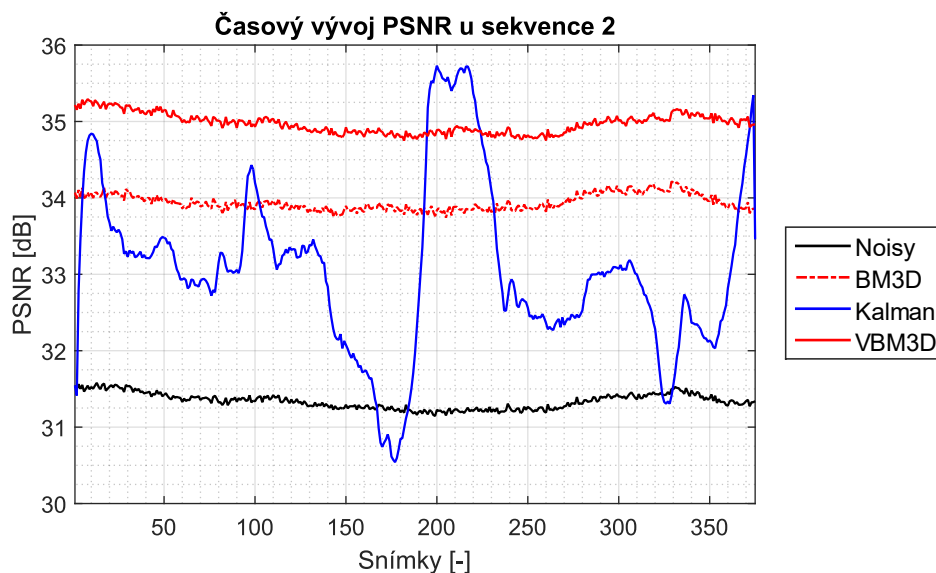


Obrázek 39: Grafy průměrných hodnot PSNR, SSIM a VIF u sekvence 1 s AWGN

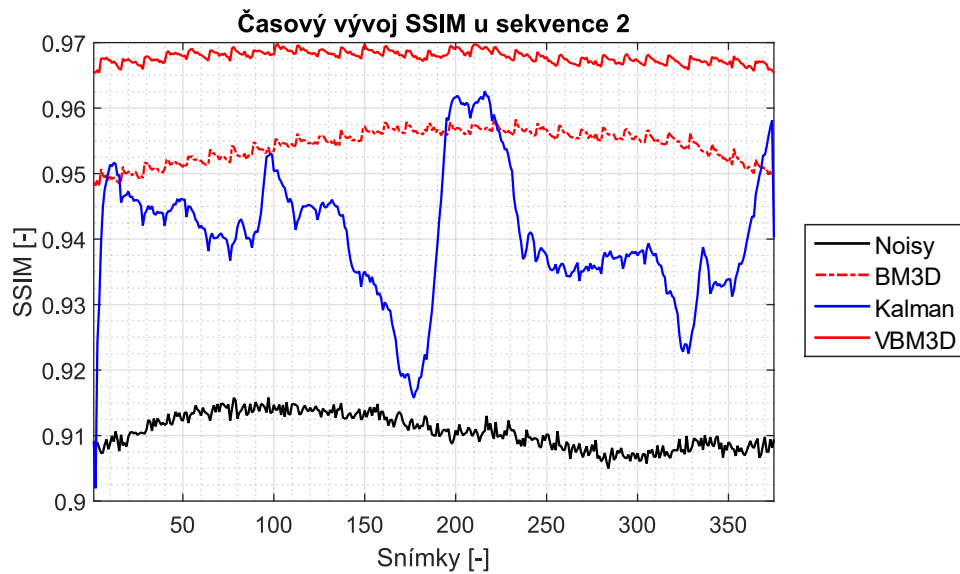
Je ale vidět, že nejpokročilejší snímkový algoritmus BM3D je i dle metriky VIF hodnocen jako lepší než degradovaná sekvence.

V dalších částech, kde budeme zkoumat vliv pohybu, se zaměříme na algoritmy pracující v časové doméně, tedy Kálmánovu filtraci a VBM3D algoritmus. Pro srovnání uvedu i jeden snímkový, a to konkrétně BM3D, aby bylo možné srovnat, jak velký vliv má časová závislost mezi jednotlivými snímky na výslednou účinnost redukce šumu.

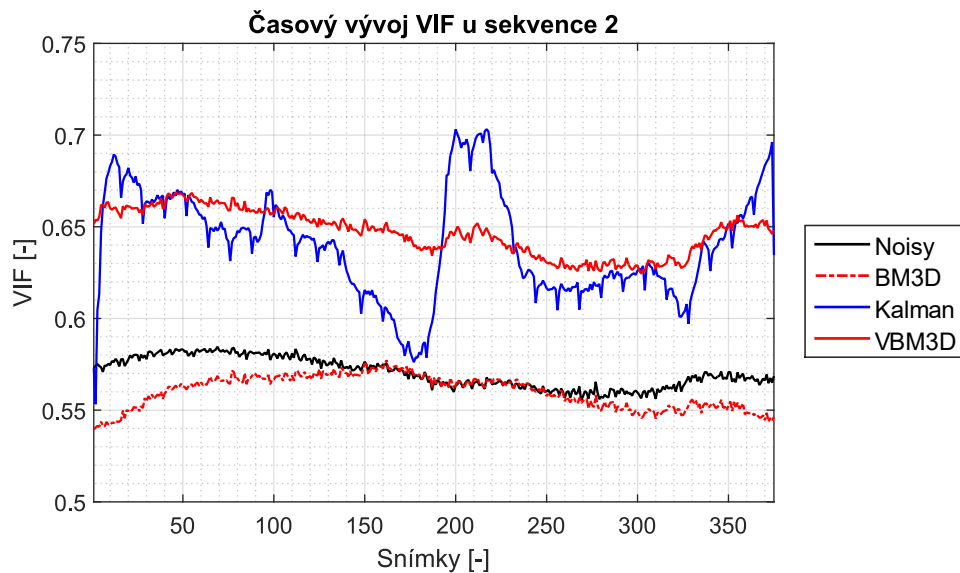
Sekvence číslo 2 zachycuje vcelku pomalý pohyb parkujícího vozidla, které se nepohybovalo konstantní rychlostí a několikrát se zcela zastavilo. Nejprve budeme opět testovat sekvenci degradovanou nekonvenčním modelem šumu. Hodnoty směrodatných odchylek šumu byly stejné jako v předchozích případech.



Obrázek 40: Graf vývoje PSNR u sekvence 2

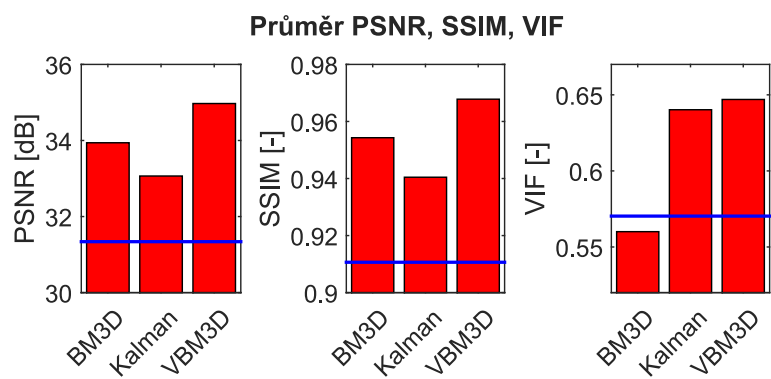


Obrázek 41: Graf vývoje SSIM u sekvence 2



Obrázek 42: Graf vývoje VIF u sekvence 2

Z grafů jsou patrné obdobné výsledky jako u sekvence 1 s tím, že Kálmánův filtr zde dosahuje o něco horších průměrných výsledků (Obrázek 43) vzhledem k tomu, že ve scéně je jistý druh pohybu stále. Jelikož je pohyb vcelku pomalý, tak nedochází k tam výraznému ghostingu. V sekvenci také výrazněji potlačuje šum, což je vidět z výsledku metriky VIF.



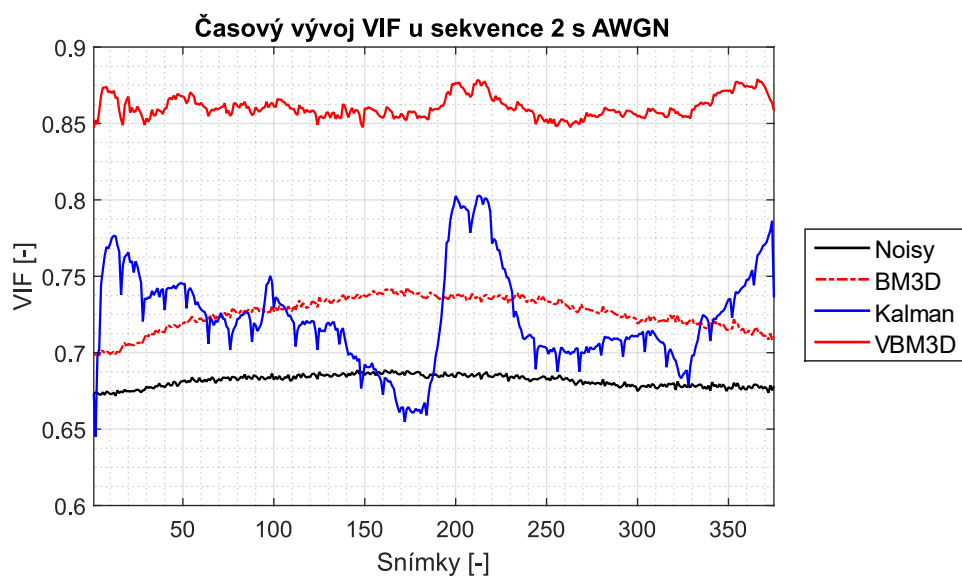
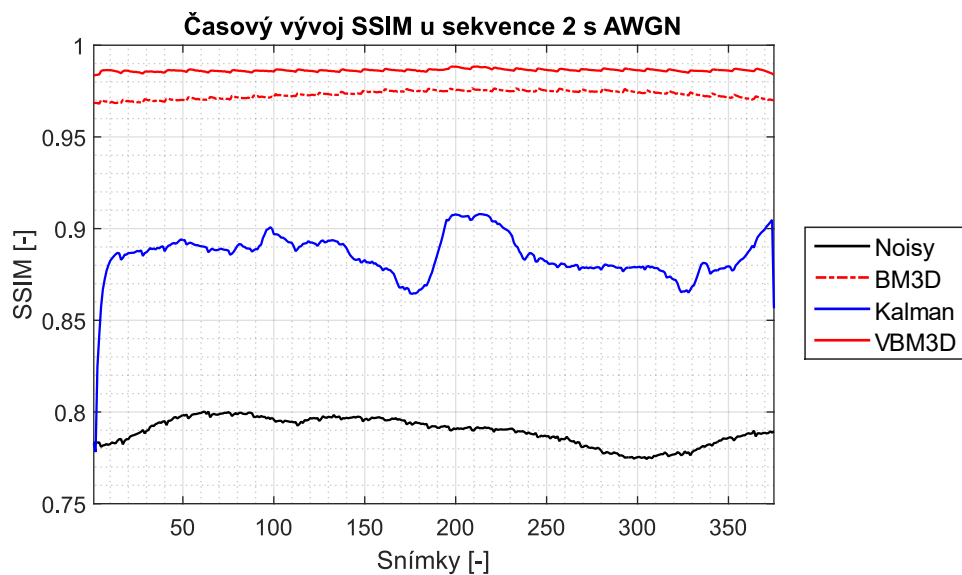
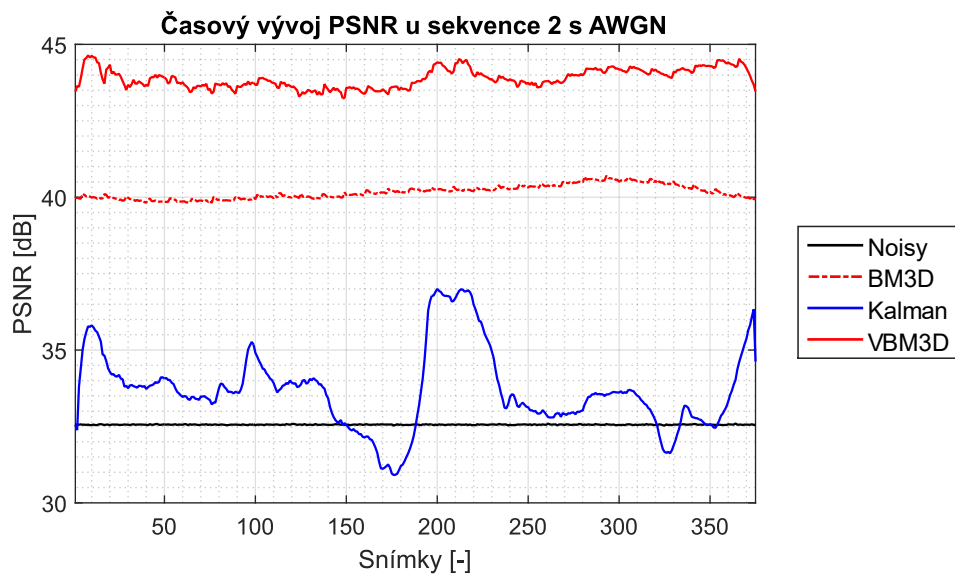
Obrázek 43: Průměrné hodnoty PSNR, SSIM a VIF u sekvence 2

Níže jsou na obrázku uvedeny ilustrační náhledy jednotlivých snímků sekvence číslo 2. Konkrétně se jedná o následující snímky: 1, 20, 50, 100, 150, 175, 200, 280 a 330.



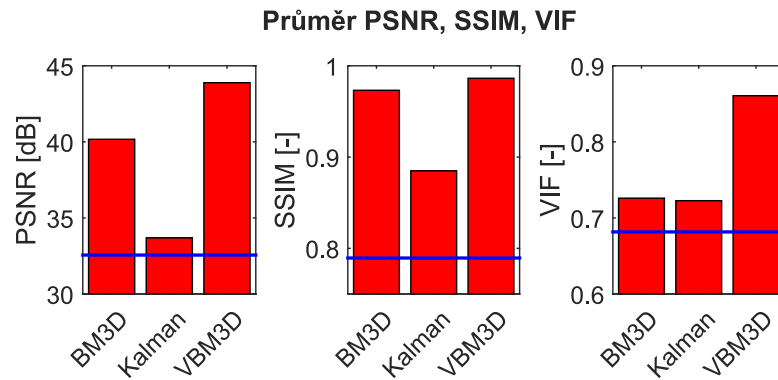
Obrázek 44: Ukázkové snímky sekvence číslo 2

Na další straně (Obrázek 45) jsou uvedeny postupně všechny tři grafy časového vývoje jednotlivých metrik u sekvence 2 degradované pomocí AWGN. Parametry směrodatné odchylky šumu jsou nastaveny stejně jako u sekvence 1 s AWGN.



Obrázek 45: Grafy vývoje jednotlivých metrik u sekvence 2 s AWGN

Výsledky hodnocení sekvence 2 s AWGN u jednotlivých metrik jsou velmi podobné, jako tomu bylo u sekvence 1 s AWGN. Pouze se mění účinnost Kálmánovy filtrace vzhledem k charakteru pohybu objektů ve scéně. Pro úplnost ještě doplňuji graf průměrných hodnot jednotlivých metrik.

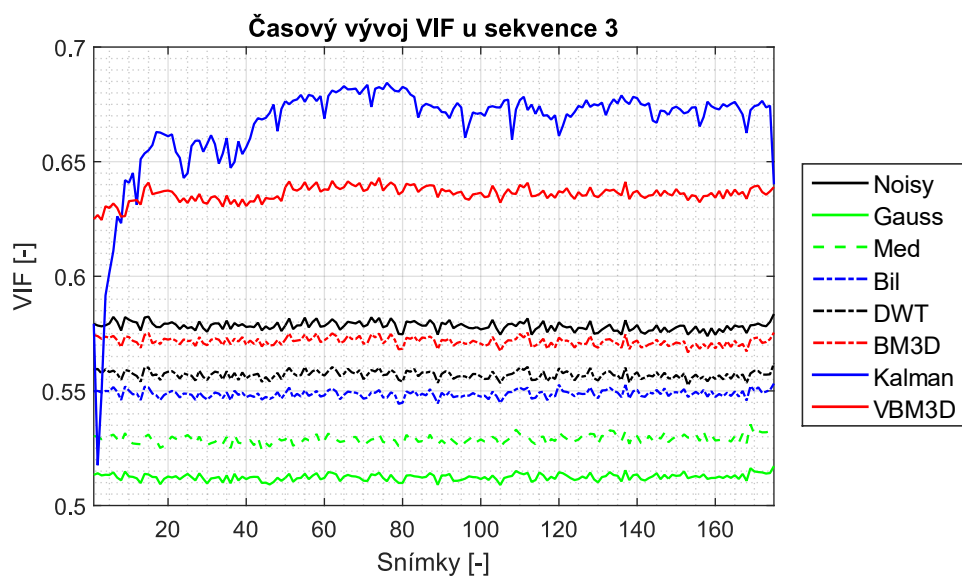
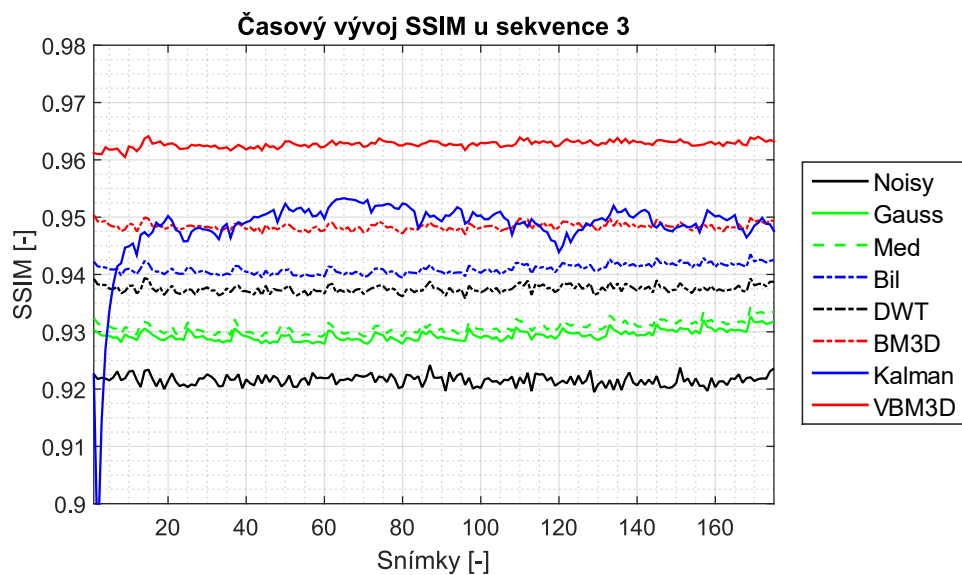
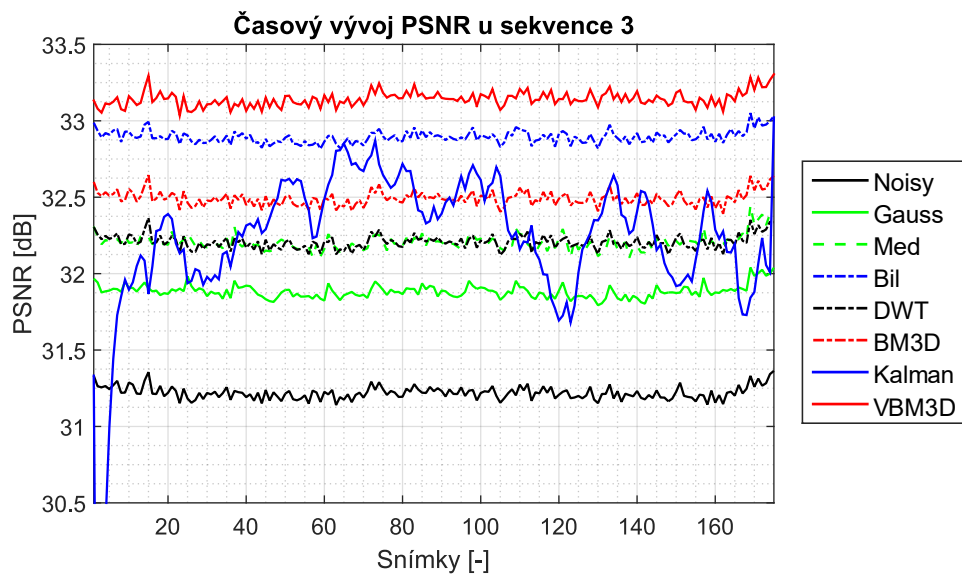


Obrázek 46: Průměrné hodnoty PSNR, SSIM a VIF u sekvence 2 s AWGN

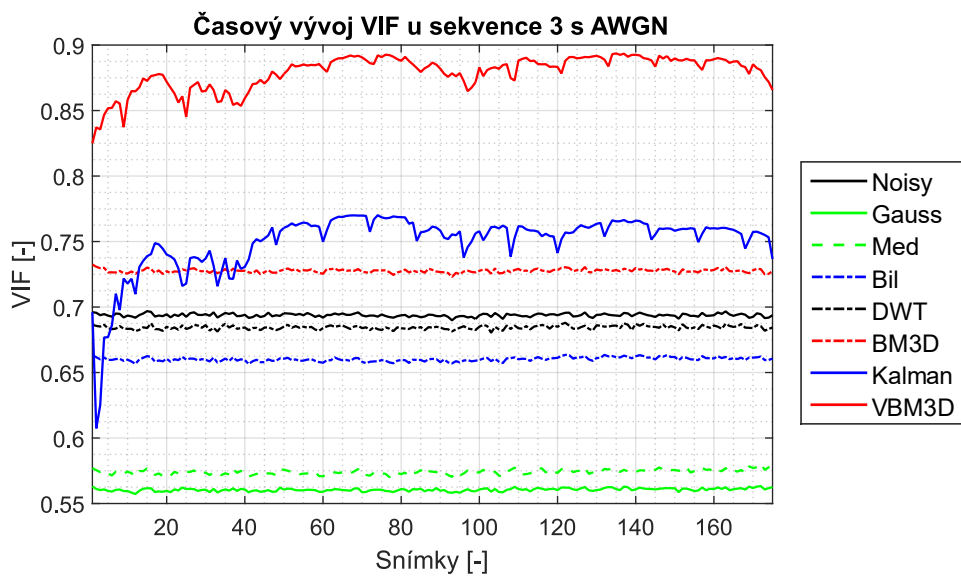
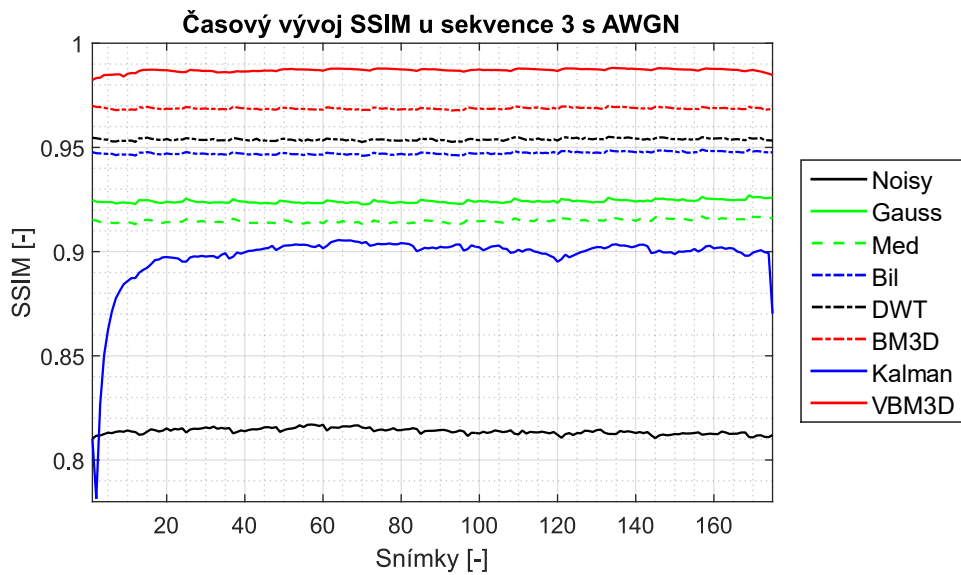
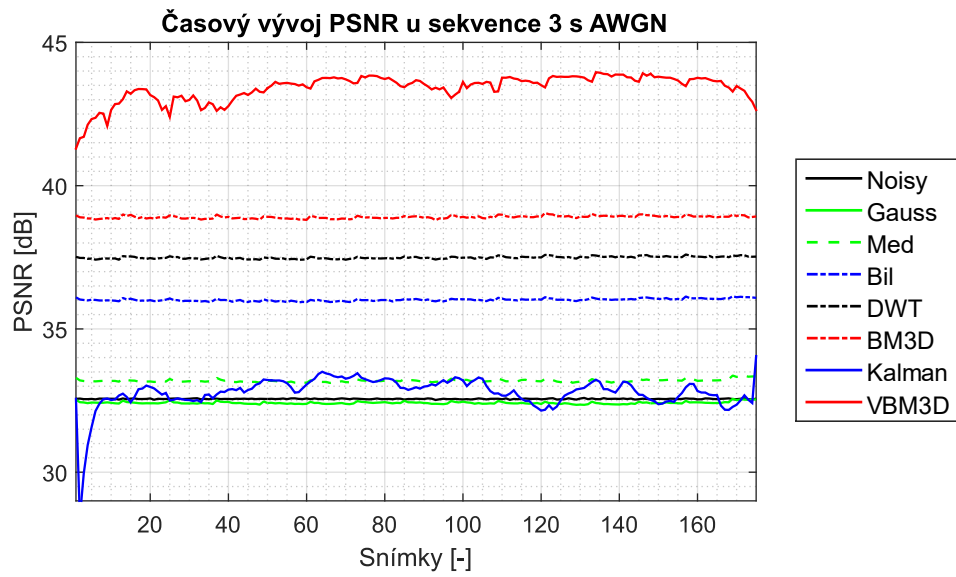
Poslední testovanou sekvencí je sekvence číslo 3, která obsahuje trochu jiný druh pohybu z hlediska tvaru a velikosti pohybujícího se objektu. V předchozích sekvencích se jednalo o automobil, který se pohyboval jako jeden větší celek. Oproti tomu u sekvence 3 jsou pohybujícími se objekty chodci. Tedy v porovnání s automobilem menší objekty, které se vlivem chůze různě mění. Tím pádem vzniká v obraze spousta nových hran a tvarů, které redukcí šumu dále komplikují. Na dalších dvou stranách proto uvedu jednotlivé časové vývoje metrik PSNR, SSIM a VIF u všech testovaných algoritmů pro sekvenci degradovanou nekonvenčním modelem šumu (Obrázek 48) a degradovanou pomocí AWGN (Obrázek 49). Níže jsou uvedeny náhledy jednotlivých snímků (1, 40, 80, 120, 160 a 175) sekvence 3.



Obrázek 47: Ukázkové snímky sekvence číslo 3



Obrázek 48: Grafy vývoje jednotlivých metrik u sekvence 3



Obrázek 49: Grafy vývoje jednotlivých metrik u sekvence 3 s AWGN

Z uvedených grafů můžeme vidět, že překvapivě Kálmánův filtr byl z hlediska účinnosti redukce šumu v tomto případě nejvíce stabilní. To je způsobeno vcelku malým a pomalu se pohybujícím objektem ve scéně. Také zde sice dochází ke ghostingu, ale pouze na malé ploše vzhledem k velikosti záběru, kde jsou ostatní části statické. Co se hran a přechodů týče, kterých zde vzniká větší množství než v předchozích případech jedoucího vozidla, je zde o něco nižší účinnost algoritmů Gaussovy a mediánové filtrace v porovnání s ostatními. V následujících tabulkách jsou uvedeny konkrétní hodnoty pro jednotlivé metriky a algoritmy. Šedou barvou je zvýrazněna hodnota vstupní sekvence degradované šumem.

Tabulka 4: Průměrné hodnoty PSNR

Průměrné hodnoty PSNR [dB] u jednotlivých metod redukce šumu						
Typ šumu	Nekonvenční model			AWGN		
Sekvence	1	2	3	1	2	3
Noisy	31,40	31,34	31,22	32,56	32,56	32,56
Gauss	32,84	33,16	31,88	33,47	33,85	32,42
Med	32,93	33,42	32,21	33,90	34,70	33,19
Bil	33,41	33,58	32,90	36,40	36,96	36,02
DWT	32,40	33,11	32,21	37,54	38,44	37,49
BM3D	32,67	33,94	32,49	38,73	40,17	38,90
Kalman	33,34	33,07	32,23	34,01	33,69	32,79
VBM3D	33,37	34,97	33,14	43,38	43,89	43,40

Tabulka 5: Průměrné hodnoty SSIM

Průměrné hodnoty SSIM [-] u jednotlivých metod redukce šumu						
Typ šumu	Nekonvenční model			AWGN		
Sekvence	1	2	3	1	2	3
Noisy	0,913	0,911	0,922	0,809	0,789	0,814
Gauss	0,929	0,941	0,929	0,925	0,935	0,924
Med	0,927	0,942	0,931	0,912	0,925	0,915
Bil	0,935	0,946	0,941	0,943	0,952	0,947
DWT	0,927	0,938	0,937	0,947	0,958	0,954
BM3D	0,937	0,954	0,948	0,960	0,973	0,969
Kalman	0,949	0,940	0,948	0,901	0,885	0,898
VBM3D	0,956	0,968	0,963	0,985	0,986	0,987

Tabulka 6: Průměrné hodnoty VIF

Průměrné hodnoty VIF [-] u jednotlivých metod redukce šumu						
Typ šumu	Nekonvenční model			AWGN		
Sekvence	1	2	3	1	2	3
Noisy	0,582	0,570	0,578	0,682	0,682	0,694
Gauss	0,525	0,527	0,512	0,572	0,578	0,561
Med	0,534	0,548	0,529	0,571	0,599	0,574
Bil	0,548	0,549	0,549	0,648	0,660	0,660
DWT	0,554	0,526	0,557	0,670	0,672	0,684
BM3D	0,566	0,560	0,572	0,706	0,726	0,728
Kalman	0,687	0,640	0,666	0,766	0,723	0,750
VBM3D	0,647	0,647	0,636	0,876	0,861	0,879

Výsledky účinnosti potlačení šumu u jednotlivých algoritmů dopadly dle očekávání. Až na tři případy byl vždy nejúčinnější algoritmus VBM3D. V těchto třech případech byl lépe vyhodnocen algoritmus na bázi Kálmánovy filtrace metrikou VIF u sekvence 1 a 3 degradované nekonvenčním modelem šumu a bilaterální filtrace metrikou PSNR u sekvence 1 degradované také nekonvenčním modelem šumu. Dle subjektivního pozorování by byl ale pravděpodobně lépe vyhodnocen algoritmus VBM3D, a to především u sekvence 1, kde Kálmánova filtrace zanechávala za pohybujícím se objektem výrazný obrys – ghosting. Redukce šumu v místech, které byly statické, byla ovšem o něco vyšší. Přestože Kálmánova filtrace jako taková není příliš často využívaná pro redukci šumu u videosekvencí, za určitých podmínek poskytuje dobré výsledky. Vliv parametru intenzity Kálmánova filtru na výslednou účinnost potlačení šumu byl sice pozorovatelný, ovšem násobně rostl ghosting za pohybujícím se objektem. Jelikož jsme se zaměřovali na vliv pohybu, nebylo by vhodné nastavit vysokou intenzitu filtru, jelikož by výsledky byly zkresleny v závislosti na velikosti ploch statických a pohyblivých. Tvar křivek zůstával přesto téměř nezměněn. Došlo tedy sice ke zvýšení účinnosti algoritmu z hlediska výsledné průměrné hodnoty, nicméně pohybující se objekt byl rozostřen tak silně, že nebyla možná jeho identifikace. Pracovali jsme tedy s výchozím nastavením algoritmu, jenž poskytovalo optimální průměrné výsledky z hlediska účinnosti potlačení šumu a schopnosti rozpoznat pohybující se objekt, byť s jistým ghostingem. Účinnost algoritmů, které nebraly v úvahu časovou závislost jednotlivých snímků, byla dle předpokladů. Na videosekvence nejsou ale vhodné, a to především z hlediska výpočetní náročnosti u bilaterální filtrace a algoritmu založeném na vlnkové transformaci. Nejúčinnější algoritmus, který byl zaměřen pouze na jednotlivé snímky, byl BM3D ve většině případů. Naopak za nejméně účinné algoritmy byly vyhodnoceny základní Gaussova a mediánová filtrace, což je výsledek, jaký jsme očekávali a testovali jsme je spíše pro ilustraci. Ne vždy se s nimi ovšem nedá dosáhnout dobrých výsledků. Jak Gaussova, tak mediánová filtrace při volbě optimálních parametrů dokáží velmi efektivně filtrovat určité druhy šumu a navíc jsou velmi jednoduché a výpočetně nenáročné. Nekonvenční model šumu, který jsme vytvářeli dle analýzy reálného systému, byl pro algoritmy vždy o něco složitější. Respektive u něj algoritmy nedosahovaly takové účinnosti jako u modelu s AWGN. Tento fakt byl pravděpodobně ovlivněn i odhadem směrodatné odchylky šumu, kterou jsme do některých algoritmů zadávali jako vstupní parametr. Sekvence

ale obsahovala šum s různou směrodatnou odchylkou v závislosti na prostorové pozici šumu, jestli se jednalo o hrany nebo jednoduté plochy ve obraze.

7.2 Výpočetní náročnost

U jednotlivých algoritmů jsme se rozhodli srovnat také jejich výpočetní náročnost. V níže uvedené tabulce jsou uvedeny průměrné časy, které byly nutné k potlačení šumu u videosekvence v rozlišení 592×992 obrazových bodů. Jednalo se o sekvenci dlouhou 10 sekund s 25 snímků za sekundu, přičemž uvedený výpočetní čas je přepočten jako průměrný na 1 snímek. Obsah scény byl obdobný jako v referenční sekvenci číslo 1 (viz Tabulka 2). Model šumu, kterým byla daná sekvence degradována, neměl pozorovatelný vliv na výpočetní náročnost. Časy jsou pouze orientační k porovnání jednotlivých metod, respektive jejich konkrétních implementací (viz kapitola 6.3). Šedou barvou jsou zvýrazněny algoritmy, které neberou v úvahu časovou závislost jednotlivých snímků, a tedy postupně ve for cyklu filtrují nezávisle každý jednotlivý snímek videosekvence. Pro test výpočetní náročnosti byl použit notebook Dell Inspiron 7568 osazený procesorem Intel Core i5 6200U, s pamětí RAM 8 GB a SSD diskem.

Tabulka 7: Výpočetní náročnost jednotlivých algoritmů

Algoritmus	Gauss	Med	Bil	DWT	BM3D	Kalman	VBM3D
čas [s]	0,011	0,013	7,28	14,96	1,29	0,023	1,28

Je vidět, že rozdíly v časech u jednotlivých algoritmů jsou poměrně značné. Jednoduchá Gaussova a mediánová filtrace jsou velmi rychlé i přesto, že se zaměřují na jednotlivé snímky videosekvence postupně a nezávisle na sobě. Další v pořadí je Kálmánova filtrace, která je také velmi rychlá a nenáročná. Je překvapivé, že algoritmus BM3D aplikovaný na jednotlivé snímky sekvence a VBM3D, který je uzpůsoben pro videosekvence a bere v úvahu časovou souvislost mezi jednotlivými snímky, jsou téměř stejně výpočetně náročné. Algoritmus VBM3D poskytuje ovšem vyšší účinnost redukce šumu (kapitola 7.1). Mezi časově nejnáročnější patří bilaterální filtrace a algoritmus na bázi vlnkové transformace. U nich byl čas nutný k výpočtu neúměrně dlouhý k tomu, jak účinně dokázaly redukovat šum. Je ale nutné podotknout, že jsme testovali pouze konkrétní implementace, které nemusejí být optimalizovány z hlediska výpočetní náročnosti.

Závěr

V této diplomové práci jsem se zabýval potlačením šumu v barevných videosekvencích. Nejprve jsem v teoretické části práce rozebral danou problematiku z hlediska toho, co je šum, jak vzniká a proč je vhodné jeho intenzitu ve videosekvencích i obecně v obraze minimalizovat. Dále jsem podal přehled současných metod pro redukci šumu od základních filtrací obrazu až po velmi pokročilé moderní algoritmy na bázi časoprostorového průměrování.

V praktické části jsem analyzoval šum u reálného snímacího systému. Pro práci jsem si zvolil bezpečnostní systém, který byl tvořen nízkorozpočtovými kamerami. Bohužel u něj nebylo možné příliš ovlivnit snímací podmínky ani kontrolovat nastavené parametry snímání. Zjistil jsem, že šum u analyzovaného bezpečnostního systému je velmi silně ovlivněn jednak kompresním algoritmem výstupního videa, tak vnitřním algoritmem na potlačení šumu, kterým disponují daná snímací zařízení. Analyzoval jsem šum za různých světelných podmínek i s infračerveným přísvitem, kterým kamery disponovaly. Jelikož se IR přísvit řídil automatikou, která jej aktivoval již za zhoršených světelných podmínek, nebylo možné analyzovat systém v noci za normálního režimu snímání. S IR přísvitem se kamera přepnula do nočního módu, ve kterém pracovala pouze jako šedotónová. Intenzita šumu byla za těchto podmínek nízká, jelikož byl přísvit silný a ve snímaném okolí je pouliční osvětlení.

Získané vlastnosti šumu jsem dále využíval v simulacích, kdy jsem vytvořil nekonvenční model šumu dle analyzovaných parametrů a následně ověřoval účinnost vybraných algoritmů pro potlačení šumu. Jednalo se celkem o sedm algoritmů, čtyři zaměřené na jednotlivé snímky bez jejich vzájemné souvislosti – Gaussovu filtraci, mediánový a bilaterální filtr, algoritmus na bázi vlnkové transformace a BM3D algoritmus a další dva beroucí v úvahu i časovou závislost jednotlivých snímků sekvence – Kálmánovu filtraci a VBM3D algoritmus. Jelikož byl šum u reálného systému silně ovlivněn kompresními algoritmy, bylo velmi ztěženo jeho modelování. Proto jsem se rozhodl využít jen určitých vlastností a vytvořit model, který by se mu přibližoval na základě vizuální podobnosti. Jako další jsem zvolil jeden z nejčastěji používaných modelů šumu u dané problematiky, a to aditivní bílý Gaussovský šum. Účinnosti vybraných algoritmů na redukci šumu jsem ověřoval pomocí několika objektivních metrik s plnou referencí. K tomuto účelu jsem nasnímal několik referenční videosekvencí kvalitní videokamerou za podmínek odpovídajících analyzovanému systému. Následně jsem je degradoval pomocí dvou vytvořených modelů šumu a vybranými algoritmy u nich šum zpětně potlačil a snažil se tak získat jejich originál. Jednotlivé testované algoritmy jsem srovnal jak z hlediska účinnosti redukce šumu, tak z hlediska jejich výpočetní náročnosti. Výsledky účinnosti jednotlivých algoritmů potvrdily předpoklady, které jsem dále rozebral. Z hlediska výpočetní náročnosti se ukázalo nevhodné používat složité filtrace obrazu pro jednotlivé snímky bez jejich vzájemné časové souvislosti. Nejlépe dopadl ve většině případů VBM3D algoritmus, což je také nejpokročilejší z vybraných algoritmů. Dále se ukázalo, že i filtr na bázi jednoduché Kálmánovy filtrace dokáže za optimálních podmínek podat kvalitní výsledky a není nikterak výpočetně náročný. Mezi algoritmy, které se zaměřovaly na jednotlivé snímky sekvence bez jejich vzájemné časové souvislosti, byl nejlépe hodnocen BM3D algoritmus,

což je obdoba VBM3D algoritmu neberoucí v úvahu časový vývoj jednotlivých snímků. Jako nejméně účinné byly hodnoceny algoritmy na bázi Gaussovy filtrace a mediánový filtr. Ty jsem ovšem testoval spíše pro demonstraci, jelikož pro daný typ šumu nejsou zcela vhodné. Neznamená to ale, že by nedokázaly dobře potlačovat určitý druh šumu. Nakonec jsem jednotlivé algoritmy srovnal ještě z hlediska jejich výpočetní náročnosti. Zde je ale nutné zdůraznit, že se jednalo o konkrétní implementace, které nemusily být z hlediska výpočetní náročnosti optimalizovány.

Pro veškerou programátorskou část práce jsem zvolil software MATLAB, kde jsem vytvořil skript s několika funkcemi, který uživateli umožňuje analyzovat vlastnosti šumu u zvolené videosekvence, následně vytvořit odpovídající model šumu a ověřit účinnosti jednotlivých algoritmů pomocí objektivních metrik s plnou referencí. V příloze jsou také obsaženy výsledky jednotlivých metrik, které skript v odpovídající podobě vykresluje. Mezi možná vylepšení bych řadil tvorbu grafického uživatelského rozhraní pro jednodušší obsluhu, rozšíření o další metriky hodnocení a metody redukce šumu. Dále sofistikovanější tvorbu modelu šumu a jeho lepší porovnání s reálným šumem z analyzovaného systému.

Literatura

- [1] R. C. GONZALEZ a R. E. WOODS, *Digital Image Processing*, Pearson, 2002.
- [2] L. WEI a L. WEISI, „Additive White Gaussian Noise Level Estimation in SVD Domain for Images,“ *IEEE Transactions on Image Processing*, sv. 22, č. 3, pp. 872-883, 2013.
- [3] H. C. RAYMOND, H. CHUANG-WA a M. NIKOLOVA, „Salt-and-Pepper Noise Removal by Median-Type Noise Detectors and Detail-Preserving Regularization,“ *IEEE Transactions on Image Processing*, sv. 14, č. 10, pp. 1479-1485, 2005.
- [4] S. W. HASINOFF, „Photon, Poisson noise,“ *Computer Vision: A Reference Guide*, pp. 608-610, 2014.
- [5] A. BOVIK, *Handbook of Image and Video Processing*, Academic Press, 2000.
- [6] M. DOSTÁL, *Využití vlnkové transformace při zpracování obrazu: diplomová práce*, Brno: Vysoké učení technické v Brně, 2015.
- [7] J. BREITKOPF, „Nekonvenční metody pro rekonstrukci obrazu ve videosekvencích,“ České vysoké učení technické v Praze, Praha, 2017.
- [8] R. BENEŠ, *Segmentace ultrazvukových snímků za účelem detekce arteriální stěny a měření vrstev intim-media: dizertační práce*, Brno: Vysoké učení technické v Brně, 2013.
- [9] E. ANISIMOVA, J. BEDNÁŘ a P. PÁTA, „Zpracování obrazu pomocí vlnkové transformace,“ *Elektro revue*, sv. 15, č. 4, pp. 238-246, 2013.
- [10] J. ŠVIHLÍK, *Aplikace DWT pro potlačení šumu v obraze*, Praha: České vysoké učení technické v Praze, 2005.
- [11] D. L. DONOHO a I. JOHNSTONE, „Ideal spatial adaptation by wavelet shrinkage,“ *Biometrika*, sv. 81, č. 3, pp. 425-455, 1994.
- [12] D. DONOHO a I. JOHNSTONE, „De-noising by soft-thresholding,“ *IEEE Transactions on Information Theory*, sv. 41, č. 3, pp. 613-627, 1995.
- [13] F. LUISER, T. BLU a M. UNSER, „A new SURE approach to Image denoising: Interscale orthonormal wavelet thresholding,“ *IEEE Transactions on Image Processing*, sv. 16, č. 3, pp. 593-606, 2007.
- [14] R. SIHAG, R. SHARMA a V. SETIA, „Wavelet Thresholding for Image De-noising,“ *IJCA Proceedings on International Conference on VLSI, Communications and Instrumentation*, pp. 20-24, 2011.
- [15] S. G. CHANG, B. YU a M. VETTERLI, „Adaptive Wavelet Thresholding for Image Denoising and Compression,“ *IEEE Transactions on Image Processing*, sv. 9, č. 9, pp. 1532-1546, 2000.
- [16] P. SHUKLA a J. SRIVAS, „Video Denoising Algorithm: A Review,“ *International Journal of Computer Science and Information Technologies*, sv. 7, č. 3, pp. 1519-1523, 2016.
- [17] G. WELCH a G. BISHOP, „An Introduction to the Kalman Filter,“ University of North Carolina at Chapel Hill Department of Computer Science, 2001.

- [18] Z. WANG a G. VARGHESE, „Video Denoising Based on a Spatiotemporal Gaussian Scale Mixture Model,“ *IEEE Transactions on Circuits and Systems for Video Technology*, sv. 20, č. 7, pp. 1032-1040, 2010.
- [19] K. DABOV, A. FOI, V. KATKOVNIK a K. EGIAZARIAN, „Image Denoising by Sparse 3-D Transform-Domain Collaborative Filtering,“ *IEEE Transactions on Image Processing*, sv. 16, č. 8, pp. 2080-2095, 2007.
- [20] K. DABOV, A. FOI a K. EGIAZARIAN, „Video denoising by sparse 3D transform-domain collaborative filtering,“ v *15th European Signal Processing Conference*, 2007.
- [21] L. HAN, G. GENG, J. XIONG a Z. XIE, „Real-color Image Denoised and Enhanced Synchronously Based on Wavelet Transform,“ v *Second International Conference on Intelligent Computation Technology and Automation*, 2009.
- [22] S. MEHER, „Color Image Denoising with Multi-channel Circular Spatial Filtering,“ v *12th International Conference on Computer Modelling and Simulation*, Orissa, 2010.
- [23] V. ABRAMOVA, „ResearchGate,“ 2015. [Online] Dostupné z: <https://www.researchgate.net/post/How_do_we_know_whether_an_image_is_affected_by_additive_noise_or_multiplicative_noise> [Stav z: 8. 1. 2018].
- [24] J. ŠVIHLÍK, K. FLIEGEL, P. KOTEN, E. JERHOTOVÁ, S. VÍTEK a P. PÁTA, „Analysis and Suppression of Noise in Astronomical Video,“ v *21st International Conference Radioelektronika*, Brno, 2011.
- [25] Z. DENGWEN a C. WENGANG, „Image denoising with an optimal threshold and neighbouring window,“ *Pattern Recognition Letters*, p. 1694–1697, 2008.
- [26] P. NDAJAH, H. KIKUCHI, M. YUKAWA, H. WATANABE a S. MURAMATSU, „SSIM Image Quality Metric for Denoised Images,“ *ADVANCES in VISUALIZATION, IMAGING and SIMULATION*, pp. 53-57, 2010.
- [27] Z. WANG, A. C. BOVIK, H. R. SHEIKH a E. P. SIMONCELLI, „Image Quality Assessment: From Error Visibility to Structural Similarity,“ *IEEE Transactions on Image Processing*, sv. 13, č. 4, pp. 600-612, 2004.
- [28] H. R. SHEIKH a A. C. BOVIK, „Image Information and Visual Quality,“ *IEEE Transactions on Image Processing*, sv. 15, č. 2, pp. 430-444, 2006.

Seznam obrázků a tabulek

Seznam obrázků

Obrázek 1: Ukázka Gaussova šumu (originál, snímek degradovaný šumem, šumové pole) ...	13
Obrázek 2: Ukázka impulsního šumu (originál, snímek degradovaný šumem, šumové pole) .	14
Obrázek 3: Ilustrace kvantizačního šumu	16
Obrázek 4: Ukázka konvoluční filtrace	18
Obrázek 5: Příklad Gaussova filtru	19
Obrázek 6: Úzkopásmový šum ve spektru.....	19
Obrázek 7: Bilaterální filtr: (a) původní signál s hranou, (b) jádro filtru, (c) výsledek filtrace (Převzato z [6]).....	21
Obrázek 8: 2D DWT dekompozice	23
Obrázek 9: Rozklad obrázku pomocí DWT	23
Obrázek 10: Dekompozice ve frekvenční oblasti	24
Obrázek 11: Ilustrace prahování: originál, tvrdé prahování, měkké prahování.....	25
Obrázek 12: Ilustrace pohybové kompenzace	29
Obrázek 13: Schéma algoritmu VBM3D	31
Obrázek 14: Kamera HIKVISION analyzovaného systému.....	32
Obrázek 15: Snímaná scéna s vyznačenými výřezy pro další testy	33
Obrázek 16: Schéma simulace reálného systému	34
Obrázek 17: Schéma separace šumového pole.....	34
Obrázek 18: Šumové pole vybraného výřezu	35
Obrázek 19: Výřez s vyznačenými vzorky pro analýzu šumových parametrů.....	36
Obrázek 20: Průměrný histogram šumu za špatných podmínek (plochy/hrany)	38
Obrázek 21: Průměrný histogram šumu za velmi špatných podmínek (plochy/hrany).....	38
Obrázek 22: Průměrný histogram šumu s IR přísivitem (plochy/hrany)	38
Obrázek 23: Ukázka nasnímané referenční sekvence	42
Obrázek 24: Schéma tvorby šumu.....	42
Obrázek 25: Maska pro zesílení šumu v oblastech hran	43
Obrázek 26: Generované šumové pole (pro lepší viditelnost s roztaženým histogramem)....	44
Obrázek 27: Detailní porovnání šumového pole (reálné/modelované)	44
Obrázek 28: Schéma objektivního hodnocení.....	49
Obrázek 29: Graf vývoje PSNR u sekvence 1	50
Obrázek 30: Ukázkové snímky sekvence číslo 1 (1, 30, 60, 90, 120 a 140).....	50
Obrázek 31: Graf vývoje SSIM u sekvence 1.....	51
Obrázek 32: Vývoj SSIM po ploše (tmavší je horší)	51
Obrázek 33: Graf vývoje VIF u sekvence 1.....	52
Obrázek 34: Ghosting u Kálmánovy filtrace	52
Obrázek 35: Grafy průměrných hodnot PSNR, SSIM a VIF u sekvence 1	53
Obrázek 36: Graf vývoje PSNR u sekvence 1 s AWGN.....	53
Obrázek 37: Graf vývoje SSIM u sekvence 1 s AWGN	54
Obrázek 38: Graf vývoje VIF u sekvence 1 s AWGN	54

Obrázek 39: Grafy průměrných hodnot PSNR, SSIM a VIF u sekvence 1 s AWGN	55
Obrázek 40: Graf vývoje PSNR u sekvence 2	55
Obrázek 41: Graf vývoje SSIM u sekvence 2	56
Obrázek 42: Graf vývoje VIF u sekvence 2	56
Obrázek 43: Průměrné hodnoty PSNR, SSIM a VIF u sekvence 2	57
Obrázek 44: Ukázkové snímky sekvence číslo 2	57
Obrázek 45: Grafy vývoje jednotlivých metrik u sekvence 2 s AWGN.....	58
Obrázek 46: Průměrné hodnoty PSNR, SSIM a VIF u sekvence 2 s AWGN.....	59
Obrázek 47: Ukázkové snímky sekvence číslo 3	59
Obrázek 48: Grafy vývoje jednotlivých metrik u sekvence 3	60
Obrázek 49: Grafy vývoje jednotlivých metrik u sekvence 3 s AWGN.....	61

Seznam tabulek

Tabulka 1: Směrodatné odchyly šumu analyzovaného systému	37
Tabulka 2: Seznam referenčních sekvencí	41
Tabulka 3: Parametry referenčních sekvencí.....	41
Tabulka 4: Průměrné hodnoty PSNR.....	62
Tabulka 5: Průměrné hodnoty SSIM	62
Tabulka 6: Průměrné hodnoty VIF	63
Tabulka 7: Výpočetní náročnost jednotlivých algoritmů	64

Přílohy

Souborová struktura datového nosiče (ZIP archiv)

- **kořenový adresář\README.txt** ...textový soubor s popisem jednotlivých souborů v ZIP archívu
- **\DATA\SEKVENCE*.mat** ...ukázkové referenční sekvence
- **\DATA*.mat** ...napočítané výsledky metrik objektivního hodnocení účinnosti algoritmů pro potlačení šumu pro dané sekvence
- **\DOKUMENTACE\BREITKOPF_DP_2018.pdf** ...tento dokument v elektronické verzi ve formátu PDF
- **\IMPLEMENTACE\BREITKOPF_DP_2018.m** ...hlavní vytvořený skript
- **\IMPLEMENTACE\FUN*.m** ...vlastní vytvořené funkce
- **\IMPLEMENTACE\FUN\PREVZATE*.zip** ...originály převzatých implementací algoritmů pro potlačení šumu a objektivních metrik
- **\IMPLEMENTACE\FUN\PREVZATE** ...6 dalších složek obsahujících převzaté funkce (některé funkce mohou být upraveny pro naše potřeby – jejich originály jsou v odpovídajících ZIP archívech)

Návod k obsluze hlavního skriptu

`\IMPLEMENTACE\BREITKOPF_DP_2018.m`

- Skript je dělen do dvou částí a několika odpovídajících sekcí
- Každá sekce nese název odpovídající její funkčnosti
- Je vhodné jednotlivé sekce spouštět postupně (například pomocí CTRL+ENTER), následně je uživatel o postupu informován pomocí příkazového okna a dalších dialogových oken
- První část je zaměřená na analýzu šumu, druhá část pak na simulaci a ověření účinnosti vybraných algoritmů pro potlačení šumu
- Druhá část začíná sekcí pro modelování šumu. Pro tuto část není nutné spouštět jakoukoli sekci z první části
- Pomocí poslední sekce lze vykreslit výsledky napočítaných metrik do grafů. (Tuto sekci lze spustit nezávisle na předchozích)